

# THE UNITED STATES OF AMERICA

**TO ALL TO WHOM THESE PRESENTS SHALL COME:**

**UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office**

**April 12, 2004**

**THIS IS TO CERTIFY THAT ANNEXED HERETO IS A TRUE COPY FROM  
THE RECORDS OF THE UNITED STATES PATENT AND TRADEMARK  
OFFICE OF THOSE PAPERS OF THE BELOW IDENTIFIED PATENT  
APPLICATION THAT MET THE REQUIREMENTS TO BE GRANTED A  
FILING DATE.**

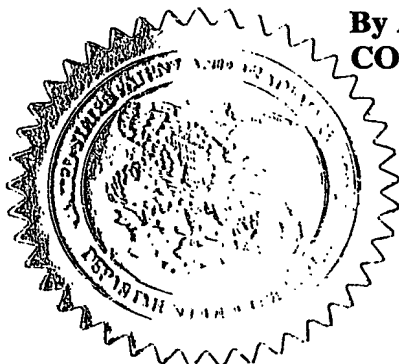
**APPLICATION NUMBER: 60/439,708**

**FILING DATE: January 13, 2003**

**RELATED PCT APPLICATION NUMBER: PCT/US04/00838**

RECEIVED	
16 APR 2004	
WIPO	PCT

**By Authority of the  
COMMISSIONER OF PATENTS AND TRADEMARKS**



*L. Edelen*

**L. EDELEN  
Certifying Officer**

**PRIORITY  
DOCUMENT**

**SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH RULE 17.1(a) OR (b)**

Best Available Copy

01-14-03

60439708

01/13/03

01/13/03  
1c355 U.S. PRO

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PTO/SB/16 (10-01)  
Approved for use through 10/31/2002. OMB 0851-0032  
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE**PROVISIONAL APPLICATION FOR PATENT COVER SHEET**

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 CFR 1.53(c).

Express Mail Label No. ET235563244US

01/13/03  
1c355 U.S. PRO  
60439708

INVENTOR(S)						
Given Name (first and middle (if any))	Family Name or Surname	Residence (City and either State or Foreign Country)				
Betty Martin Joseph	Bird Yablonski Laurenzano	New York, NY	New York, NY	New York, NY	New York, NY	New York, NY
<input type="checkbox"/> Additional inventors are being named on the _____ separately numbered sheets attached hereto						
TITLE OF THE INVENTION (500 characters max)						
CONSUMER CARE MANAGEMENT METHOD AND SYSTEM						
Direct all correspondence to: CORRESPONDENCE ADDRESS <input type="checkbox"/> Customer Number _____ <span style="margin-left: 50px;">Type Customer Number here</span> <span style="border: 1px solid black; padding: 5px; float: right;">Place Customer Number Bar Code Label here</span> OR						
<input checked="" type="checkbox"/> Firm or Individual Name	Marc E. Hankin, Esq.,					
Address	Gordon & Rees, LLP					
Address	300 S. Grand Ave., Suite 2075					
City	Los Angeles	State	CA	ZIP	90071	
Country	US	Telephone	213-576-5006	Fax	213-680-4470	
ENCLOSED APPLICATION PARTS (check all that apply)						
<input checked="" type="checkbox"/> Specification	Number of Pages	259	<input type="checkbox"/> CD(s), Number			
<input checked="" type="checkbox"/> Drawing(s)	Number of Sheets	1	<input type="checkbox"/> Other (specify)			
<input type="checkbox"/> Application Data Sheet	See 37 CFR 1.76					
METHOD OF PAYMENT OF FILING FEES FOR THIS PROVISIONAL APPLICATION FOR PATENT						
<input type="checkbox"/>	Applicant claims small entity status. See 37 CFR 1.27.					FILING FEE AMOUNT (\$)
<input checked="" type="checkbox"/>	A check or money order is enclosed to cover the filing fees					38,908
<input checked="" type="checkbox"/>	The Commissioner is hereby authorized to charge filing fees or credit any overpayment to Deposit Account Number					
<input type="checkbox"/>	Payment by credit card. Form PTO-2038 is attached.					
<input type="checkbox"/> The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.						501990
<input checked="" type="checkbox"/> No.						
<input type="checkbox"/> Yes, the name of the U.S. Government agency and the Government contract number are: _____						\$160.00

Respectfully submitted,

SIGNATURE

Marc E. Hankin

Date 1/13/03

REGISTRATION NO.  
(if appropriate)  
Docket Number:

38,908

LTHS 1013081

TYPED or PRINTED NAME

Marc E. Hankin

TELEPHONE

(213) 576-5006

**USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT**

This collection of information is required by 37 CFR 1.51. The information is used by the public to file (and by the PTO to process) a provisional application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 8 hours to complete, including gathering, preparing, and submitting the complete provisional application to the PTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, Washington, D.C. 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Box Provisional Application, Assistant Commissioner for Patents, Washington, D.C.

60439708.0113

**CERTIFICATE OF MAILING BY "EXPRESS MAIL" (37 CFR 1.10)**Applicant(s): **BETTY BIRD, ET AL.**

Docket No.

**LTHS 1013081**

Serial No.

Filing Date

Examiner

Group Art Unit

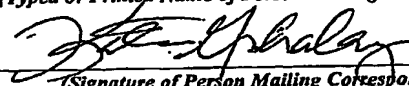
Invention: **CONSUMER CARE MANAGEMENT METHOD AND MANUFACTURE**

I hereby certify that the following correspondence:

**PROVISIONAL APPLICATION FOR PATENT COVER SHEET; PROVISIONAL PATENT APPLICATION OF  
BETTY BIRD, MARTIN S. YABLONSKI, JOSEPH R. LAURENZANO FOR US PATENT ON CONSUMER CARE  
MANAGEMENT METHOD AND SYSTEM**

*(Identify type of correspondence)*

Is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under  
37 CFR 1.10 in an envelope addressed to: The Assistant Commissioner for Patents, Washington, D.C. 20231 on

**JANUARY 13, 2003***(Date)***KIERSTEN MISHALANY***(Typed or Printed Name of Person Mailing Correspondence)*

*(Signature of Person Mailing Correspondence)***ET235565944US***("Express Mail" Mailing Label Number)***Note: Each paper must have its own certificate of mailing.**

PROVISIONAL

PATENT

APPLICATION

of

BETTY BIRD

MARTIN S. YABLONSKI

JOSEPH R. LAURENZANO

For

UNITED STATES PATENT

on

CONSUMER CARE MANAGEMENT METHOD AND SYSTEM

Docket No.: LTHS 1013081

Sheets of Drawings: 1

Gordon & Rees, LLP  
300 South Grand Avenue, Suite 2075  
Los Angeles, California 90071  
(213) 576-5000



This is an application for a Provisional Patent pursuant to 35 USC § 111 (b)

Consumer Care Management Method and System

by

Betty Bird,

Martin S. Yablonski,

Joseph R. Laurenzano

BACKGROUND

The number of people in need of vision rehabilitation services is growing exponentially-particularly, as the older adult population continues to live longer. People who are blind or partially sighted often require a variety of vision rehabilitation services. A specialized low vision eye exam to determine the extent of vision loss, the prescription of and training in the use of optical devices to help maximize remaining vision, and services to promote independent and productive living (such as orientation and mobility (safe travel) skills, personal grooming, cooking safely, etc.) need to be monitored, adjusted and assessed according to a standard of service delivery.

While the health care field has for years worked within a universally recognized standard of service delivery, outcomes, and quality assurance, the vision rehabilitation field has never developed a similar standard. There has been no way to concretely track the success of various vision rehabilitation therapies, collect a repository of data, and follow a trajectory for the individual receiving services. This type of information is valuable for people with vision loss and their families, as well as for service providers like the Lighthouse International (herein "Lighthouse") and other organizations.

Clearly there is a need for a standardized method to concretely track the success of various vision rehabilitation therapies, collect a repository of data, and follow a

trajectory for the individual receiving services (herein "consumer") in the field of vision rehabilitation.

## SUMMARY

5

The innovative Electronic Vision Rehabilitation Record software program and system (herein "EVRR"), designed and implemented by Lighthouse International, will be shared through web technology with vision rehabilitation organizations throughout the New York area and the country. The database will constitute the first national database for benchmarking and setting standards in the field of vision rehabilitation. The EVRR will also have national implications for people in need of services, and the reimbursements that they are entitled to from managed care organizations and Medicare.

10

It is an object of the present invention to function as an appointment scheduler accessible to all service provider members and manage resources.

15

It is another object of the present invention to provide templates of predetermined clinical pathways and related care by supplying detailed functional questions in each assessment area (currently Independent Living, Mobility, Psychotherapy, Employment, Computer Training, Academic Skills, Social Services, Low Vision).

20

It is a further object of the present invention to provide uniform templates for the elements of each component, such as assessments, plans, for comparability and future expansion.

It is yet another object of the present invention to standardize service quality through uniform assessments and analyses.

25

It is a further object of the present invention to manage interviews through scripts, prompts and reminders.

It is another object of the present invention to provide consent to treat and HIPAA requirements.

It is a further object of the present invention to provide a clinical pathway approach with levels of service intensity.

30

It is still another object of the present invention to provide a service plan based on consumer functional goals.

It is a further object of the present invention to record progress by goal and objective, not open unstructured narratives.

It is another object of the present invention to report through structured records with consistency.

5 It is a further object of the present invention to provide pre- and post- service assessments that have been conducted.

It is still another object of the present invention to measure functional improvement.

10 It is a further object of the present invention to provide routine periodic reports. These reports may include numbers of referrals, consumers, hours, service provider, visits, average hours per consumer, cost per consumer, revenue per consumer, revenue per service provider by consumer, by hour.

It is another object of the present invention to provide a query capability available to users at different levels.

15 It is a further object of the present invention to serve as a reporting data warehouse with thirty-six rolling months of information for trend analysis and year-over-year reporting.

It is yet another object of the present invention to provide the average percent improvement by consumer, intervention, and service provider.

20 It is a further object of the present invention to determine and provide quality benchmarks and variances/exceptions reports.

It is another object of the present invention to bill from the appointment and/or visit information as well as bill electronically for external parties equipped to receive electronic transmission.

25 It is a further object of the present invention to provide corrective actions.

It is still another object of the present invention to provide security, including limiting access to information would be restricted by service provider role and "need to know" in compliance with HIPAA.

30 The EVRR addresses standard Triage Questions and Prompts. The EVRR enables a professional service provider member to ask the individual with vision loss a series of standard questions, moving through a series of system actions and prompts that trigger

recommendations to a particular clinical pathway: i.e. mobility (safe travel) therapy, psychotherapy, and activities of daily living. Once the individual agrees to the recommendations within the various specialized areas, appointments will be made for further assessment, to be followed by appointments for the actual therapy if applicable.

5       The EVRR addresses appointment and resource scheduling. The EVRR contains a fully accessible appointment scheduling system, including systems accessible to blind or partially sighted persons using speech and large print computer output. Currently, appointments for individuals must be scheduled on either a one-time or recurring basis. EVRR supports the scheduling of appointments at any location. The EVRR enables  
10   appointments to be made in a convenient manner for consumers (grouping appointments together on the same day, or sequencing them so as to maximize learning), as well as support the most efficient scheduling of service provider, equipment and facility resources.

      The EVRR addresses support of various payment methods. Most of the payers  
15   for services are moving to an outcomes-based payment model in which payment is made only upon the documented achievement of an outcome (i.e. achieving independent mobility within the home). Other payers and programs still reimburse on a fee-for-service basis, or use a voucher model, where a payer-issued voucher must be returned to the payer with documentation that the service was completed. Further, payers such as  
20   Medicare use electronic payment whereby billing data is transmitted via a modem connection. The EVRR supports all of these payment methods.

      The EVRR addresses payer fulfillment. Payer requirement fulfillment is designed so that outputs can easily be changed to match payer's coverage levels and documentation requirements.

25       The EVRR addresses billing requirements. Most vision rehabilitation centers have extensive medical billing requirements. The EVRR is flexible enough to meet the needs of all of these sources.

      The EVRR addresses quality assurance and consumer satisfaction. The EVRR collects quality assurance and satisfaction data, based on both its own methodologies and  
30   industry standards. The EVRR produces various reports for managerial review, and

downloads for further analysis using the program evaluation departments' statistical analysis software.

5 The EVRR addresses reporting requirements. The EVRR enables time spent by service providers and other resources to be tracked for each individual for billing and reporting.

After a consumer undergoes an initial triage and specialized assessments, s/he is classified into a specific level of care prescribed by the agency. For instance, a consumer with minimum needs may only require brief low vision services, while a consumer with complex needs may require more lengthy low vision as well as psychotherapy and/or  
10 other services.

The consumer profiles extend from initial clinical pathway templates that include an Initial Assessment (Level 0), Consumer With Minimum Restorative Needs (Level 1), Consumer With Moderate Restorative Needs (Level 2), and Consumer with Complex Restorative Needs (Level 3). However, sometimes the level of care must be changed  
15 after treatment begins. There are several reasons why such changes may be necessary. In these instances, the EVRR indicates this as either a variance (changed or subsequently undetectable of condition) or an error (clinician oversight, protocol variance, imperfect protocol).

Additional advantages of the present invention will be specifically set forth by the  
20 following description as are obvious from the description and use of the present invention. The advantages of the present invention will be realized and attained by means of the elements and combinations particularly pointed out in the appended claims. Therefore, the foregoing general and detailed descriptions are exemplary and explanatory only and should not restrict the present invention as claimed.

25

#### BRIEF DESCRIPTION OF DRAWINGS

FIG. 1, is a block diagram of a customer order entry and service arrangement using the EVRR system from the service provider's point of view.

30

## BEST MODE FOR CARRYING OUT THE INVENTION

5 The preferred embodiment of use for the innovative EVRR system will be shared through web technology with vision rehabilitation organizations at a minimum throughout the New York area as well as the United States. The preferred embodiment further comprises a plurality of computers connecting, by secure Internet connections, to a central computer. The EVRR performs program analysis, evaluation, tracking, standardization of care, scheduling of appointments, and coordination of billing and payment procedures, and reduction of administrative costs.

10

## SYSTEM HARDWARE AND OPERATION

15 According to the invention, at least one central computer and associated system and stored consumer data is accessible by several other remote service provider computers. The service provider has at least two choices to use the computers and system. The order in which a service provider may use the computers and associated system when providing care to a consumer will illustrate the interaction between the computers and associated system.

20 Considering now the remote computers in greater detail, a service provider computer includes at minimum a processor with a primary memory unit, such as a random access memory, a monitor with connecting video cable, a keyboard with a connecting keyboard cable. The remote computers may be located on desk tops of service providers, bedside to consumers, in nurse stations, in offices, in laboratories or any other location which facilitates providing care to the consumers as well as portable  
25 ("laptop") computers. EVRR is installed onto each remote computer. The service provider computer must be of suitable computing speed and memory capacity. Additionally, the service provider computer must have an Ethernet connection or a 56K modem as well as a CD Rom drive.

30 Considering now the interaction between the service provider and consumer in reference to the service provider remote computer and system, the service provider logs onto either a desktop computer or portable ("laptop") computer with EVRR installed and

affiliated system. Data may be captured directly into the computer of the service provider and stored within a database of the service provider computer.

Periodically, the service provider is required to connect to the central computer via Internet through Ethernet or a modem. The Internet connection enables a data  
5 exchange between the service provider computer and the central computer. The Internet connection to the central computer also provides other services, such as calendar updates, standard reporting, software updates, and other services addressed herein.

The second choice the service provider has to use the computers and system comprises the service provider remote computer to always be connected to the central  
10 computer via a high speed Internet connection. A direct exchange of information between the service provider's computer and the central computer takes place when the service provider chooses to connect to the central computer in this manner.

Considering now the central computer, the central computer comprises at least one server, preferably a plurality of servers, at Lighthouse in New York. The central  
15 computer is capable of servicing a plurality of remote service provider computers individually and selectively. The central computer and associated system store templates, consumer information, store and adjusts clinical pathways for data entered, store and adjusts care and/or training information, store and adjusts follow-up information, store and analyzes success rate of care and/or training given and other functions addressed  
20 herein.

Considering now the central computer in greater detail, the central computer includes at minimum a processor with a primary memory unit with a connecting memory cable, a monitor with connecting video cable, a keyboard with a connecting keyboard cable. Preferably, the central computer comprises a server farm of suitable computing  
25 speed and memory capacity. The central computer is horizontally and vertically scaleable. The central computer provides security, including security within HIPAA compliance for both captured and transmitted information. Data is stored both on site in the Lighthouse servers as well as offsite. Additionally, data is processed in a data warehouse for statistical analysis.

30 The arrangement is a hardware and software system or a hardware, software and internet system which facilitate communication of such information as consumer

information, care and treatment information, social services information and feedback information regarding the foregoing from the use of the system.

## SYSTEM AND SOFTWARE OPERATION

5

Users of the EVRR and system will be trained to use the EVRR and system. As noted earlier, the arrangement is a software and hardware system that facilitates the communication, analysis, billing, tracking, and care of consumer data and related care data. Although all individual consumer or localized consumer group data and related  
10 care data may be used without an intranet, the preferred embodiment uses an Internet based intranet to make full use of all of the EVRR and system capabilities discussed herein. In this regard, the arrangement of the preferred embodiment may be adapted to be on-at all times. All aspects of the isolated or semi-isolated (non-Internet) use as well as use comprising Internet connection, is included in the preferred embodiment as well as  
15 other aspects and uses will be discussed.

According to the present invention, in each stage, the EVRR and system are used to enter consumer data and care data FIG. 1. Predetermined templates of clinical pathways and related pathways are accessed and updated as the data entered increases. Three levels, or templates for clinical pathways, exist to determine the consumer's  
20 restorative needs. These pathways include minimum, moderate and complex levels of patient need and may entail job site analysis, consumer and employer interviews, occupational research, informational interviews, development of a vocational goal, identification of educational, rehabilitation and vocational training needs, testing for employability skills. Calendar appointments and other information may be uploaded  
25 from the central computer by the service provider at any time, but for illustration purposes, the service provider logs onto the central computer to access the daily schedule 1 before seeing any customers 2.

Following an initial Registration 3 and Recording 4 of the customer, an initial assessment is preformed by a Social Worker, Low Vision Doctor, Low Vision Therapist  
30 or Assistant in order to determine the consumer's vision restorative needs to maximize existing vision. In this assessment, a Comprehensive Low Vision examination, including



detailed history, functional assessment, and testing (including refraction), prescribing, dispensing and instruction in the use of low vision devices is accomplished. This entire assessment process is termed "Triage" 5. Two levels, or templates for clinical pathways, exist to determine the consumer's restorative needs for use of functional vision and may involve instruction in the use of Low Vision devices, a specialized evaluation and training for the more complex needs of the low vision, education in the use of or very high powered and/or specialized devices, as well as counseling, rehabilitation and further medical attention assessments 6. Following Triage, answers to questions and/or data are saved onto the service provider's computer and the different clinical pathways and services, including such services as billing, are determined, as discussed below and represented as 7 in FIG. 1.

An assessment by a Certified Social Worker with two years experience or a licensed Psychologist may be preformed executed to develop treatment plan for individual referred for psychotherapy due to vision loss resulting from disease or trauma. 15 This may include the administration of the various testing tools, the identification of symptomatology for indicators of treatment, and a psychiatric evaluation to determine if medication or hospitalization would be appropriate. If medication is prescribed, follow-up psychiatric evaluation of medication will be conducted as directed by the payer. Individual can be re-referred for services when stabilized, and will be reassessed by a 20 CSW or licensed Psychologist. Three levels, or templates for clinical pathways, exist to determine the consumer's restorative needs. These pathways include minimum, moderate and complex levels of patent need. The treatments may include varying duration and intensity of psychotherapy.

An assessment may be preformed done by a Certified Rehabilitation Teacher, 25 VRA or RTA in order to determine the consumer's vision restorative needs to establish a baseline of current functioning, needs and responsibilities of daily life skills, develop treatment plan maximize existing vision. Three levels, or templates for clinical pathways, exist to determine the consumer's restorative needs for independent living needs. These pathways include minimum, moderate and complex levels of patent need 30 comprising simple instruction to intense training of such living skills as safety, self care

basis (personal care, eating, clock, phone, writing lists, checks, letters, Braille, and computers) or special needs such as cognitive or physical long term training.

5 A Certified Orientation and Mobility Specialist may an assessment in order to determine assessment of skills and needs, development of goals, and development of a treatment plan to restore skill level lost due to vision impairment. In this assessment, three levels, or templates for clinical pathways, exist to determine the consumer's restorative needs. These pathways include minimum, moderate and complex levels of patent need and may involve instruction in agreed upon and needed skills to restore function.

10 An assessment of skills, needs, and recommendation for equipment and training assessment by a Assistive Technology Specialist or Technology Instructor may be preformed in order to assess present skills, access needs, system needs, and to recommend appropriate equipment and training. Several classifications, or templates, exist within determining the consumer's computer training needs. Computer instruction may include instruction in print enhancing programs, word processing and spreadsheet  
15 programs, screen reading software, keyboard layout, the Internet, and email.

An assessment of consumer education, work history, skills and job interests may be preformed by an Employment Specialist in order to determine job readiness and/or barriers to employment for unemployed individuals, or to assess the modifications and accommodations necessary to enable an employed individual to remain in their job.

20 These assessment results are in a defined pathway level that includes the goals, objectives and hours of service for a particular consumer. Hours of service are tracked via the Visit Record. Visits are comprised of various data elements, namely the date; duration, description, and payment source for each service provided. When necessary, the visit record may also contain a diagnosis (ICD) and treatment (CPT) code. Some or  
25 all of these items may be required by external entities for billing purposes. Visits should be recorded after each session.

The EVRR also comprises Visit Record templates for use with various interventions. These forms are meant to be attached to a Incremental Record of Activity (herein "IRA"), so that both the visit and goal attainment data are together in one, easy to  
30 read document. The EVRR will deduct each visit duration from the stipulated number of hours for the consumers' service level, thereby tracking how much time has been used

and what remains. This tracking enables the service provider to gauge the consumers' progress and determine if the hours remaining will be sufficient to achieve all outstanding goals. If not, providers may request additional hours if they believe this will lead to a successful outcome.

5        Once clinical pathways and level of care criteria are in place, clinical outcomes can begin to be collected 8. A sample of data items for low vision could include items addressing improvement in vision acuity and self-reported satisfaction with the ability to perform certain activities or for psychotherapy, sample data could include items addressing improvement in status and self-reported satisfaction with the ability to  
10    function in society.

Consumer data is collected at four points in time. The first is at the assessment stage, addressing the consumer's experiences, difficulties in performing activities, interests and needs, and the like. The second is at the point when the intervention ends and the provider marks all goals for which the consumer agreed to receive service, re-  
15    rates various items s/he originally rated during the assessment, and completes a questionnaire concerning the consumers' level of participation in the learning process. The consumer also re-rates each "self-assessment" item on which they originally stated they had a great deal of difficulty performing, and for which they received training. Thus, a "post-service" score is derived for comparison with the "pre-service" assessment  
20    score. The third is at the end of the Episode, addressing the consumers' assessment of and satisfaction with the their services and providers. Finally, the fourth is at a point six months after the Episode ends, concerning the consumer's experiences during the period after the completion of treatment. The consumer is also asked the same "self-assessment" questions they originally completed during their assessment and later when they  
25    completed their training.

In each instance where the consumer is asked to provide feedback, the surveys are completed independently by the consumer or with the assistance of a third party (*not* the service provider). These surveys are conducted on a random sample of individuals. Eventually, it may be helpful to extend certain measurement processes beyond six  
30    months in order to quantify whether any benefits are discernible over the longer term.

Providers will be prompted to complete a standard set of QA questions, regardless of the intervention or service level, when all goals for their intervention have been marked. However, no QA questions will be used when a consumer was assessed only and did not receive further service. Also, each item that was scored a 5 or below during the assessment (using the 1-10 Provider scale) is re-rated.

The Consumer Self-Assessment Survey consists of all Assessment items in which consumer rated him/herself a 1 or 2 (on a scale of 1 - 4), and for which the consumer agreed to receive training. Thus, the Consumer Feedback is Intervention specific, but not Level specific. Responses will be compared with the information originally collected in the Assessment to determine if the consumer believes they have made progress and are more capable of performing tasks/implementing what they learned. This data can also be compared with the intervention outcome to learn if providers and consumers have different views of service delivery. These items are unique for each consumer, and each intervention.

The Six Month Post Episode Survey consists of two components: a repetition of the consumer self-assessment items originally completed during the assessment process (and therefore intervention specific), and a more general survey. Both components are aimed at identifying former consumers who may require additional services. Consumers are randomly selected to participate, and as with the End of Episode QA, are interviewed by a third party, not the service provider.

Many network members possess the financial management skills to estimate costs per unit of service for low vision and psychotherapy. Certain members also possess the skills to estimate the average cost per consumer. However, very few members can estimate how the cost per consumer differs for a minimum, moderate or complex level individual. For the following reasons, it is critically important that agencies understand how the cost of care differs for consumers at different intensity levels. With this in mind, the EVRR agencies to utilize standard cost accounting protocols in order to estimate costs per unit and costs per consumer for each of the intensity levels described by the clinical pathways.

The service provider, after servicing the consumer, may either have additional consumers or may have serviced the final consumer of the day 9. From time to time or

after the service provider has seen and/or preformed services for customers, the service provider is required to connect to the central computer via Internet through Ethernet or a modem 10. As stated above, the Internet connection enables a data exchange between the service provider computer and the central computer and accomplishes such tasks as calendar updates, standard reporting, updates and other data manipulations. The Internet connection to the central computer also provides other services, such as calendar updates, standard reporting, software updates, and other services addressed herein. If the service provider connects to the central computer after servicing all customers for that day, the service provider has successfully completed his/her day 11.

10

## SYSTEM SOFTWARE OPERATION

Starting the program will bring up the welcome menu. All records are accessed through the Search Menu. The first two command buttons are for users of speech software. Throughout the system, command buttons are placed by text messages that cannot be read by Jaws and other screen readers. Clicking on these will repeat the text that is either above or to the left of the buttons.

15

ENTER A NEW CONSUMER: SEARCH

Click on the Search Menu Button or keystrokes alt e on the first screen to enter the system and arrive at the search criteria screen. To be certain the consumer doesn't already have a case, enter the last name and select search (Alt e). To use the Clear Screen key, Alt I, to blank out any information and refresh the fields.

20

If a consumer is not found, a blank search result screen will appear. The search result consists of a header section containing the information about the selected consumer. The lower portion of the screen is a grid that can contain consumer(s) meeting the search criteria entered on the previous screen, such as service provider name or intervention. In the case of a new consumer, the grid is blank.

25

SEARCH AND WORK WITH EXISTING CONSUMERS

Consumers can be retrieved from the main search menu by selecting from the any of the available fields. To work with a specific individual, type in the last name and then search. To retrieve those records assigned to an individual service provider member, use

30

the dropdown list to select the service provider member and then search. Similarly, searching can be done on a particular Intervention. For example, a search is done for all consumers referred to Employment. If the search results shows consumers in Employment, intervention code 050. The header record gives the total number of  
 5 consumers matching the search criteria and positions the record indicator on the first consumer, sorted in alphabetic order by last name in a grid view below the header record.

There are three ways to retrieve records. Either use Alt N to move to the next consumer, the next consumer button or the mouse to click on the desired consumer. The VIP name fields will change to match the selection. Once the consumer is identified, use  
 10 the main menu function button, Alt M, to access all records for that consumer. In this example, the grid row becomes highlighted when using the mouse to select a consumer, the name matches the selected row, and the current consumer counter is now less than before.

The simplest way to enter data in the system is to type the value in the current  
 15 field and then tab to the next field. Assign an agency standard case number (the system will prompt if the case number already exists), as case numbers are not automatically generated. Throughout the system there are triggers that will either move the cursor to a required field, skip a section completely or automatically fill in information. This will happen once the user tabs past the key field. The first such trigger  
 20 happens on page 1 of registration. For example, if the answer to question d, relationship to person that is visually impaired is self, then section e, VIP name, will auto populate. The cursor intentionally remains on the VIP First Name field to allow speech software to remain focused.

A dropdown window is a field containing a down arrow. This indicates that there  
 25 are coded values to be chosen. A blank dropdown field only allows for one choice while a dropdown displaying a list, such as the last two fields on the Basic Information 1 screen supports multiple values.

There are several ways to select from a blank dropdown list. The simplest is to type the first letter or number if known. In the case of Yes/No values a y or n will  
 30 populate the field. An alternate method is to click on the down arrow using the mouse and the selecting the value. A third way is to use the up and down arrow keys to scroll

through the values. To remove an erroneous entry, tab to the field and use the backspace key.

To select from a multiple value dropdown list, either use the mouse and click whatever values are appropriate or, arrow down the list and use the spacebar to select the  
5 codes. Selected codes will appear in dark blue.

To remove an unwanted value, use the spacebar or the mouse.

Tabbing on the last field in page one of registration will automatically turn to page two. Alt n is also the equivalent keystroke. Conversely, Alt p moves the cursor to a previous page. This can also be accomplished by using the mouse and clicking on the  
10 respective page headings on the top of the form.

To complete the registration process, fill in all demographics and enter the service provider of the registrar and record the registration date. Registration date is a critical field for the continuity of service to move the consumer into the triage phase.

Use the save key or alt s to save the record. Every time a save is done, the system  
15 presents a verification that the record has been saved. To end registration, close the screen using the close button or Alt C.

Throughout the system there are comment fields, which may contain instructions or information notes. These areas are not accessible to screen reader software. To allow blind workers to have access to the text, there is a button to the right of the narrative area.  
20 Embedded in the button is a popup window that displays the same text in an accessible format. Use the Esc key to clear the message.

Once Registration has been completed and saved, the triage portion of the system may be worked on if the consumer is reselected from the search screen. Once registered, the search results screen now has the Main Menu button activated.

## 25 MAIN MENU:

The Main Menu has buttons that lead to all parts of the consumer's record. If a consumer has just registered, no modules are available except for Registration, Triage and Visual & Health Information. Until Triage is completed, the consumer entering the system has not started an episode of service.

30

**TRIAGE SCREENS:**

Triage consists of a series of seven screens designed identify problem areas, recommend interventions and record consumer agreement of these recommendations. In addition to the seven main triage screens there is also a form for entering relevant data for the federal statistical form, 7-OB. Triage questions identify problem areas using the most direct means of questioning. The computer will skip over questions that seem redundant or unnecessary.

**INTERVENTION RECOMMENDATIONS:**

During triage, the computer generates the intervention recommendations. These are based upon the consumer response to status and problem identification questions. Not all questions have to be answered. If a recommendation for a particular intervention is triggered by the answer to a question, the computer has a built in skip pattern that will pass over the remaining questions that would trigger that intervention. The following example demonstrates the recommendation process. Answering "Yes" to a visual functioning problem and "Yes" to a household activities task on Page C, triggers two recommendations for interventions on Page F, Recommendations, seen on the next page. It is assumed that the subsequent triage screens are filled in and saved- A through E. In order for the recommended interventions to become active in the system, the consumer must accept each intervention.

Additional recommendations can be added manually here if other services are requested. Then, the triage phase must close with a close date on Page G, Filing Triage. Without a closure date, the intervention records will not be released and assessments functions will not be available. Episode Closure date is included on this screen should the consumer reject all services being offered. At the present time, it is also the only place to close an episode.

Visual & health information can be entered at anytime during an episode of service, but it is strongly recommended that this data be entered at the beginning of the case, during the triage phase. It is available to workers at the beginning of each assessment so this information is a valuable resource. More than one of these records can exist over time and can be entered on a periodic basis. The most current record will be presented in the assessments. Enter, save and close this screen upon completion. In each



of the assessments, a button is available to bring up the most current visual and health data.

Whenever multiple records exist, there will be fields at the top of the screen to indicate the total number of records that exists for the consumer and the sequence of the record being viewed. In the following screen example, there are two visual and health records for the consumer. On the bottom of forms where multiple records can be stored, there are also navigation buttons for next record (Alt X) and previous record (Alt V) movement.

#### PAYERS

10 Payer information in the Prototype system is informational only. Once triage is completed and the consumer is ready for assessments, the payer button on the consumer's main menu becomes available. Payer information is stored by each occurrence of an intervention. As additional interventions are added to the consumers record during the service delivery phase, so to additional payer data can be added.

#### 15 ENTERING PAYER:

This is a layered screen and is a little more involved since the desired result is to attach payers to interventions where there may be a single payer with multiple interventions and also multiple authorization numbers. Follow these steps when entering a payer.

- 20 1. Triage must be completed, dated and saved.
2. When on the payer screen, create a payer record by selecting new payer. (Alt W and enter).
3. Fill in payer information on the top portion of the screen. Also enter the consumer ID and any payer contact information.
- 25 4. Enter authorization information for the first intervention. You cannot save the screen without an authorization/intervention record. If no authorization is required, enter "none" in the Authorization Number field. If Interventions do not appear in the drop down list, then triage is not completed. Exit to the main menu, reselect triage and save it with a close date.
- 30 5. Save the record (Alt V).

6. To add another authorization record with intervention to the same payer, select New Authorizations (Alt A).

7. To move on to the next new payer, select new payer and continue with the same process.

5        Once triage is completed, the main menu will have additional buttons activated. For example, Interventions and Assessments functions are available. Interventions are the programs that were accepted in triage. There are also additional interventions that do not have assessments associated with them. Viewing interventions, making referrals to non triage based interventions and entering visit information follows this section.

10       There are five computerized assessments that stem from the triage recommendations. Assessments set the goals, objectives and levels of service for the consumer. On the screen, the Interim Record of Activity button is grayed out. Only after at least one assessment has been completed can any progress reporting and goal achievement be recorded in the IRA.

15       Enter the assessment menu by clicking on the assessment button or by using Alt A. All Five assessments have common computerized elements which will be discussed. For the specifics of each of the assessments, please reference the manual forms.

The function keys to access the assessments are:

Alt M Mobility,  
20    Alt I Independent Living,  
      Alt C Computer Skills,  
      Alt E Employment,  
      Alt S Social Service.

25       Following the same design structure of registration and triage, the assessments are composed of a series of tabs, each labeled in alphabetical order.

There are numerous dropdown boxes which do not have to be used to fill in the values. Numeric scores can be manually entered and it is better if tabbing and typing becomes the norm. As in triage, narrative fields are often followed by a highlighted box which contains the script for speech users. Alt N and Alt P will navigate between pages.  
30    Alt S will save a form.

Goals and Objectives are automatically generated by the system based upon the rating and response to the series of goal and objective questions found in each of the assessments. Scoring is also done by the computer based upon the consumer and provider ratings and will give a recommended level of service. Before a score can be calculated, the preceding screens should be saved.

On the screen displaying the level (different screen depending upon the assessment, but always found following the learning strategy and factors section), the provider has the opportunity to change the level of service and also record the consumer's agreed upon level of service.

The last page of each assessment is the service agreement page, which will summarize the agreed upon goals. The learning strategy and the factors that influence training must be completed before going to the service agreement page. The service agreement page contains goals that have been previously been identified. Additional fields are available to record any equipment or other services. Learning Strategies & Factors must be completed before goals will appear on the service agreement. Once completed, dated and saved, a service agreement can be printed out using either the print button or Alt R. The report uses whatever the default computer is on the workstation.

Whenever a user leaves an assessment that has not been completed, the system will ask if the assessment is completed. Answering "No" exits the assessment. On the other hand, answering "Yes" will prompt the user to enter a file date, the completion date of the assessment. This is a critical point in the generation of goals and objectives. It is the dating and filing of the assessment that moves the goals and objectives into the IRA. It is an important reminder for ongoing service delivery.

#### SOCIAL SERVICE ASSESSMENT:

This assessment has a unique navigation feature, based upon the social needs and the risk factors identified in triage. On page A of the assessment, Historical Information, the last field in the form is a button Assess Social Service (s). Use that button to go the appropriate pages that address the risk factors. Do not use the next and previous page buttons or alt key combinations. This will break the logical page skip pattern. By using the Access Social Service(s) button, and then either tabbing or using the next page button the EVRR will navigate to the appropriate pages. At first this can cause some confusion,

but with practice, this function will prove to be a considerable timesaver. Only use the next and previous page buttons when reviewing the social service assessment or if adding additional information not included in section 9, the triggers.

- 5 There is a corresponding IRA for each assessment that has been completed. To access the IRA menu, navigate from the consumer main menu by either clicking on the IRA button or using Alt D. As other assessments are completed, their corresponding buttons will be activated.

The function keys to access the IRAs are:

- Alt M Mobility,
- 10 Alt I Independent Living,
- Alt C Computer Skills,
- Alt E Employment,
- Alt S Social Service.

All IRAs have the same format.

- 15 1) The header section lists the consumer information, episode number, intervention identifier and record number. This is protected and is informational only.

2) Progress Report Information identifies the service provider author of the report, in most cases, the person assigned to provide the service in the intervention, the report period covered and any narrative information

- 20 3) Goal Information contains a primary area for entering goal achievements and achievement dates and a grid view showing the goals.

4) Objective Information contains a similar structure to goals. For the goal selected those corresponding objectives are displayed.

#### ENTERING GOALS AND OBJECTIVES:

- 25 Goals are composed of one or more objectives. Before a goal can be completed, all objectives should have their achievements recorded. Navigate between goals by either using the mouse and clicking on the selected goal or by using the functions Alt L (next goal) or Alt G (previous goal). The objectives will change to match the correct goal. To record the achievement for a particular objective, either click in the grid on the desired
- 30 objective or use Alt E (next objective) or Alt O (previous objective). Select the achievement value from the dropdown list, record the achievement date and save the

record. You must save each objective and goal achievement individually. Use Alt S to save.

Interventions equate to the programs offered by your organization and function similar to a record on a college transcript. Key fields in the intervention record are start date, end date, primary service provider assigned and outcomes. Interventions generated by triage have the start date being the date the consumer accepted the recommendation on triage Tab F, Recommendations. All other fields must be entered on the Intervention Screen. The Intervention screen serves multiple purposes; viewing and editing, entering visit records, generating referrals to additional interventions and closing and recording the end of service report.

#### VIEWING AND EDITING INTERVENTIONS:

Select the Intervention button from the consumer main menu. If the button is grayed out, there are no interventions recorded for the consumer. The intervention screen shows one intervention per page. The heading lines show the total number of interventions in the episode, the consumer identifiers and the name of the intervention. Once an intervention is active, the primary service provider member should be entered on this screen and saved. This is a dropdown list built by the system administrator. Without this service provider assignment, searching and retrieving case loads by service provider is not possible.

In order to view other interventions, use Alt N or Alt P to change to the next or previous intervention record. Reference the counters at the top of the screen as a guide when moving between interventions. The narrative block is available for entering any specifics about the intervention as it relates to the consumer. It is an open field that can be edited or added to at any time during service.

#### 25 VISITS:

Visits are the record of activity for the consumers in the intervention. There is a Visit button on the bottom of the intervention record. To enter a visit for a particular intervention, the intervention screen must be on that record. In the following example, a visit is being entered for Independent Living ("IL"):

30 1) Locate the record for IL, use Alt V to open the visit record, then

- 2) Enter the visit information (always check to heading information to be sure that it is the correct consumer in the correct intervention), and
- 3) Save and Close the screen when done.

#### INTERVENTION REFERRALS:

- 5        At any time during service delivery, additional interventions can be added to the consumer's record. The third button on the bottom of the intervention screen performs this function.

10        The system will not open an intervention that is already active and will generate a warning. However, it does allow a second instance of an intervention that has already been completed. For example, a consumer was triaged, had mobility and was closed from that intervention but is now back for more service. Another instance of mobility can be opened. Because mobility has an assessment associated with it, a new assessment must be completed.

#### END OF SERVICE:

- 15        When an intervention is completed and the related IRAs done, the outcome of the intervention and the ending date of the intervention must be entered to effectively close the service. The EVRR system has built in functionality to do a post service evaluation based upon goals and objectives in each intervention.

#### END OF SERVICE PROVIDER REPORT:

- 20        An End of Service Report for the Intervention should be done. The provider should complete the fields based on the intervention provided.

#### EPISODES

- 25        An episode can last a month, a year or several years. The trigger for creating episodes is triage. A consumer with one triage record is in Episode 1. Upon returning and a new triage record is done, the consumer is now in episode 2. The EVRR automatically creates and tracks episodes. It associates all related records, assessments, interventions, IRAs, and similar items as addressed herein, to a specific episode number.

#### ENDING AN EPISODE OF SERVICE:

- 30        When all interventions are completed and there is no more service to be given, an episode ends. To close an episode, use triage screen G. The last field on the form is episode close date. Enter the date and save the record. The New Episode Button, Alt W,

when doing a second triage. Also Previous and Next Episode buttons will activate when 2 or more triage records are present.

- 5 If there are open interventions or incomplete IRAs, an error message will be generated when attempting to enter an episode end date. The episode cannot be completed until the error is corrected by either going into interventions and closing all interventions or completing the IRAs and closing the interventions. The summary field on the IRA must contain some information.

#### POST EPISODE QUESTIONS AND ANSWERS:

- 10 There are two End of Episode screens that record consumer responses to service related questions. End of Episode QA should be done once the episode has closed. The 6 Month Post Episode QA should be completed around six months after the episode has ended for the consumer. "Microsoft" "Access" can generate a list of consumers meeting the criteria.)

#### END OF EPISODE QA:

- 15 Once an episode has closed, the end of Episode Button will be activated on the consumer main menu. Answer all questions and save the screen. The overall score which shows both pre and post service results can now be viewed for the consumer.

- 20 All function keys use the combination of Alt plus a letter (not case sensitive). First press and hold down the Alt key then strike the letter. The following list displays only the letter and description, assuming alt is the prefix.

#### GLOBAL FUNCTIONS:

- S - Save
- C - Close
- W - New Record
- 25 X - Next Record
- N - Next Page
- P - Previous Page
- E - Search

#### SEARCH RESULTS SCREEN:

- 30 M - Main Menu
- W - New Consumer

- P - Previous Consumer
- N - Next Consumer
- C - Close
- CONSUMER MAIN MENU:
- 5 R - Registration
- T - Triage
- P - Payer
- V - Visual Health
- D - IRA
- 10 A - Assessments
- I - Interventions
- Q - 6 Months Post QA
- O - End of Episode OA
- E - Search
- 15 C - Close
- REGISTRATION:
- N - Next Page
- P - Previous Page
- S - Save
- 20 C - Close
- TRIAGE:
- N - New Episode
- V - Previous Episode
- X - Next Episode
- 25 N - Next Page
- P - Previous Page
- S - Save
- C - Close
- PAYER:
- 30 P - Previous Authorization
- N - Next Authorization



- A - New Authorization
- E - Previous Payer
- R - Next Payer
- W - New Payer
- 5 S - Save
- C - Close
- VISUAL AND HEALTH INFORMATION
- W - New Record
- V - Previous Record
- 10 X - Next Record
- S - Save Page
- C - Close Page
- ASSESSMENTS:
- R - Print Agreement
- 15 W - New Assessment
- X - Next Record
- V - Previous Record
- N - Next Page
- P - Previous Page
- 20 S - Save
- C - Close
- INTERVENTIONS:
- V - Visits
- E - End of Service Report
- 25 I - Intervention Referral
- N - Next Intervention
- P - Previous Intervention
- S - Save
- C - Close
- 30 VISITS:
- W - New Visit

N - Next Visit

P - Previous Visit

S - Save

C - Close

5 IRA:

X - Next Record

V - Previous Record

E - Next Objective

O - Previous Objective

10 L - Next Goal

G - Previous Goal

S - Save

C - Close

INTERVENTION REFERRAL:

15 S - Save

C - Close

END OF SERVICE REPORT:

S - Save

C - Close

20 END OF EPISODE QA:

S - Save

C - Close

SIX MONTHS END OF EPISODE QA:

S - Save

25 C - Close

# COMPUTER CODE

30

```
Private Sub cmdClose_Click()  
    Unload Me  
End Sub
```

```
Private Sub cmdComputerSkills_Click()  
    frmComputer.Show vbModal  
End Sub
```

```
Private Sub cmdEmployment_Click()  
    frmEmployment.Show vbModal  
End Sub
```

```
Private Sub cmdIndependentLiving_Click()  
    frmILAssessment.Show vbModal  
End Sub
```

```
Private Sub cmdMobilityAssessment_Click()  
    frmMobility.Show vbModal  
End Sub
```

```
Private Sub cmdSocialWork_Click()  
    frmSocialWork.Show vbModal  
End Sub
```

```
Private Sub Form_Load()  
    Dim rst As New ADODB.Recordset
```

```
    rst.Open "Select ConsumerID From Interventions Where ConsumerID=" & CustomerID & " And Episode=" &  
    & - Episode & " And InterventionID='010' And CaseNumber = " & CaseNumber & "'", cnn, _  
    adOpenForwardOnly, adLockReadOnly  
    cmdSocialWork.Enabled = IIf(rst.EOF, False, True)  
    rst.Close
```

```
    rst.Open "Select ConsumerID From Interventions Where ConsumerID=" & CustomerID & " And Episode=" &  
    & - Episode & " And InterventionID='014' And CaseNumber = " & CaseNumber & "'", cnn, _  
    adOpenForwardOnly, adLockReadOnly  
    cmdMobilityAssessment.Enabled = IIf(rst.EOF, False, True)  
    rst.Close
```

```
    rst.Open "Select ConsumerID From Interventions Where ConsumerID=" & CustomerID & " And Episode=" &  
    & - Episode & " And InterventionID='013' And CaseNumber = " & CaseNumber & "'", cnn, _  
    adOpenForwardOnly, adLockReadOnly  
    cmdIndependentLiving.Enabled = IIf(rst.EOF, False, True)  
    rst.Close
```

```
    rst.Open "Select ConsumerID From Interventions Where ConsumerID=" & CustomerID & " And Episode=" &  
    & - Episode & " And InterventionID='050' And CaseNumber = " & CaseNumber & "'", cnn, _  
    adOpenForwardOnly, adLockReadOnly  
    cmdEmployment.Enabled = IIf(rst.EOF, False, True)  
    rst.Close
```

```
    rst.Open "Select ConsumerID From Interventions Where ConsumerID=" & CustomerID & " And Episode=" &  
    & - Episode & " And InterventionID='160' And CaseNumber = " & CaseNumber & "'", cnn, _  
    adOpenForwardOnly, adLockReadOnly  
    cmdComputerSkills.Enabled = IIf(rst.EOF, False, True)  
    rst.Close  
    Set rst = Nothing  
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Set frmAssessments = Nothing  
End Sub
```

```

Option Explicit
Dim Str As String
Dim response As String
Private rstComputer As ADODB.Recordset
Private rst As ADODB.Recordset
Private IsGoalsClicked As Boolean
Public IsNewRecord As Boolean ' Global variable for the SaveDataToDB procedure
Public IsFileDateChecked As Boolean

Private Sub cboAgencyServices_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboAgencyServices(Index).ListIndex = -1
    End If
End Sub

Private Sub cboAgree_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboAgree(Index).ListIndex = -1
    End If
End Sub

Private Sub cboAgree_LostFocus(Index As Integer)
    If Index = 11 Then
        If CheckForTab Then
            tabComputer.Tab = 9
            chkRecommend(2).SetFocus
            frmComputer.Caption = "Goals 2"
        End If
    ElseIf Index = 22 Then
        If CheckForTab Then
            tabComputer.Tab = 10
            chkRecommend(5).SetFocus
            frmComputer.Caption = "Goals 3"
        End If
    ElseIf Index = 35 Then
        If CheckForTab Then
            tabComputer.Tab = 11
            cmdLearning.SetFocus
            frmComputer.Caption = "Learning Strategy and Factors"
        End If
    End If
End Sub

Private Sub cboAgree_Validate(Index As Integer, Cancel As Boolean)
    ' If consumer answers "No" to a goal then objectives are not allowed
    If Index = 0 Or Index = 1 Or Index = 2 Or Index = 3 Or Index = 4 Or Index = 5 Or Index = 6 Or Index = 7 Then
        If cboYesNo(6).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 8 Or Index = 9 Or Index = 10 Or Index = 11 Then
        If cboYesNo(7).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 12 Or Index = 13 Or Index = 14 Then
        If cboYesNo(8).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 15 Or Index = 16 Or Index = 17 Then
        If cboYesNo(9).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 18 Or Index = 19 Or Index = 20 Or Index = 21 Or Index = 22 Then
        If cboYesNo(10).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 23 Or Index = 24 Or Index = 25 Or Index = 26 Or Index = 27 Then
        If cboYesNo(11).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 28 Or Index = 29 Or Index = 30 Or Index = 31 Or Index = 32 Or Index = 33 Or Index = 34 Or Index = 35 Then
        If cboYesNo(12).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    End If

```

End Sub

Private Sub cboEnhancingName\_KeyUp(KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboEnhancingName.ListIndex = -1

    End If

End Sub

Private Sub cboEquipment\_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboEquipment(Index).ListIndex = -1

    End If

End Sub

Private Sub cboExternalServices\_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboExternalServices(Index).ListIndex = -1

    End If

End Sub

Private Sub cboFactors\_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboFactors(Index).ListIndex = -1

    End If

End Sub

Private Sub cboKeyboardSkill\_Click()

    If cboKeyboardSkill.ListIndex <> -1 Then

        If cboKeyboardSkill.ListIndex >= 0 And cboKeyboardSkill.ListIndex < 2 Then

            chkRecommend(0).Value = 1

        End If

    End If

End Sub

Private Sub cboKeyboardSkill\_KeyUp(KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboKeyboardSkill.ListIndex = -1

    End If

End Sub

Private Sub cboLocateKeys\_Click()

    If (cboLocateKeys.ListIndex >= 0 And cboLocateKeys.ListIndex < 4) Then

        If Key identi

    ation is 1-4 then recommend Goal #1 (Keyboarding)

        chkRecommend(0).Value = 1

    End If

End Sub

Private Sub cboMagnification\_KeyUp(KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboMagnification.ListIndex = -1

    End If

End Sub

Private Sub cboModality\_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboModality(Index).ListIndex = -1

    End If

End Sub

Private Sub cboNotes\_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboNotes(Index).ListIndex = -1

    End If

End Sub

Private Sub cboOperatingSystem\_KeyUp(KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then

        cboOperatingSystem.ListIndex = -1

    End If

End Sub

```
Private Sub cboPointSize_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboPointSize(Index).ListIndex = -1
    End If
End Sub

Private Sub cboRateYourTyping_Click()
    If cboRateYourTyping.ListIndex <> -1 Then
        If cboRateYourTyping.ListIndex >= 0 And cboRateYourTyping.ListIndex < 2 Then
            chkRecommend(0).Value = 1
        End If
    End If
End Sub

Private Sub cboRateYourTyping_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboRateYourTyping.ListIndex = -1
    End If
End Sub

Private Sub cboSequencing_Click()
    If cboSequencing.ListIndex >= 0 And cboSequencing <= 4 Then
        chkRecommend(0).Value = 1
    End If
End Sub

Private Sub cboSequencing_LostFocus()
    If CheckForTab Then
        cmdTestRule.SetFocus
    End If
End Sub

Private Sub cboServiceLevel_Click(Index As Integer)
    If cboServiceLevel(0).ListIndex <> -1 Then
        cboTimeFrame.ListIndex = cboServiceLevel(0).ListIndex
        txtAssessment(27) = cboTimeFrame.Text
    End If
End Sub

Private Sub cboServiceLevel_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboServiceLevel(Index).ListIndex = -1
    End If
End Sub

Private Sub cboServiceLevel_LostFocus(Index As Integer)
    If cboServiceLevel(0).ListIndex <> -1 Then
        cboTimeFrame.ListIndex = cboServiceLevel(0).ListIndex
        txtAssessment(27) = cboTimeFrame.Text
    End If
End Sub

Private Sub cboSpeech_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboSpeech.ListIndex = -1
    End If
End Sub

Private Sub cboSpelling_Click()
    If cboSpelling.ListIndex <> -1 Then
        If cboSpelling.ListIndex >= 0 And cboSpelling.ListIndex < 2 Then
            chkRecommend(0).Value = 1
        End If
    End If
End Sub

Private Sub cboSpelling_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboSpelling.ListIndex = -1
    End If
End Sub
```

```
End If
End Sub
```

```
Private Sub cboSpelling_LostFocus()
    If CheckForTab Then
        tabComputer.Tab = 3
        cmdSpelling.SetFocus
        frmComputer.Caption = "Standardized Tests 1"
    End If
End Sub
```

```
Private Sub cboStaffs_GotFocus()
    frmComputer.Caption = "Self Assessment 1"
End Sub
```

```
Private Sub cboStaffs_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboStaffs.ListIndex = -1
    End If
End Sub
```

```
Private Sub cboStrategy_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboStrategy.ListIndex = -1
    End If
End Sub
```

```
Private Sub cboTypingSpeed_Click()
    If cboTypingSpeed.ListIndex = 0 Then
        chkRecommend(0).Value = 1
    End If
End Sub
```

```
Private Sub cboTypingSpeed_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboTypingSpeed.ListIndex = -1
    End If
End Sub
```

```
Private Sub cboUseComputer_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboUseComputer.ListIndex = -1
    End If
End Sub
```

```
Private Sub cboVisualFields_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboVisualFields.ListIndex = -1
    End If
End Sub
```

```
Private Sub cboYesNo_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboYesNo(Index).ListIndex = -1
    End If
End Sub
```

```
Private Sub cboYesNo_LostFocus(Index As Integer)
    If Index = 4 Then
        If CheckForTab Then
            tabComputer.Tab = 2
            cboHighestEduc.SetFocus
            frmComputer.Caption = "Self Assessment 3"
        End If
    End If
    If Index = 5 Then
        If cboYesNo(5).ListIndex = 1 Then
            cboKeyboardSkill.SetFocus
        End If
    End If
End Sub
```

End Sub

```
Private Sub cmdAccess_Click()  
MsgBox "Access Mode - Testing for font size needs with print enhancing software" & vbNewLine &  
For large print users: Use the hard copy sheet (pre-designed for this test). Have consumer r  
aloud each sentence beginning with the smallest print they feel they can see. Note the small  
size print line the consumer can read without any errors. Consumer should use his or her gla  
s.", vbOKOnly, "Instructions for Access Mode"  
RefreshJaws  
tabComputer.Tab = 4  
cboPointSize(0).SetFocus  
End Sub
```

Private Sub cmdClear\_Click()

```
IsNewRecord = True  
clearData cboStaffs, False  
clearData txtAssessment, True  
clearData cboVisualFields, False  
clearData cboYesNo, True  
clearData lstDevices, False  
clearData lstOtherConditions, False  
clearData lstAffectTraining, False  
clearData cboUseComputer, False  
clearData cboOperatingSystem, False  
clearData lstTypeTraining, False  
clearData cboKeyboardSkill, False  
clearData cboRateYourTyping, False  
clearData cboSpelling, False  
clearData cboLocateKeys, False  
clearData cboTypingSpeed, False  
clearData cboPointSize, True  
clearData cboEnhancingName, False  
clearData cboMagnification, False  
clearData cboSpeech, False  
clearData cboModality, True  
clearData cboNoteSpeed1, False  
clearData cboNoteSpeed2, False  
clearData cboNoteAccuracy1, False  
clearData cboNoteAccuracy2, False  
clearData cboNotes, True  
clearData cboStrategy, False  
clearData cboFactors, True  
clearData cboAgree, True  
clearData cboEquipment, True  
clearData cboAgencyServices, True  
clearData cboExternalServices, True
```

```
txtAssessment(0) = Date  
txtCurrentRecord = ""
```

```
cmdSave.Enabled = True  
tabComputer.Tab = 0
```

End Sub

```
Private Sub cmdClose_Click()  
Unload Me  
End Sub
```

```
Private Sub cmdEDoc1_Click()  
MsgBox "Perform the same test using the electronic version of the point size document. Info  
e consumer that this test is being repeated to determine if there is any difference between  
size needed in hard copy (reflected light) to print size needed in electronic format (proje  
light}. Consumer should use his or her glasses.", vbOKOnly, "Instructions for Electronic Do  
t 1"  
RefreshJaws  
tabComputer.Tab = 4  
cboPointSize(2).SetFocus  
End Sub
```



```
Private Sub cmdEDoc1_LostFocus()  
tabComputer.Tab = 4  
cboPointSize(2).SetFocus  
End Sub
```

```
Private Sub cmdEDoc2_Click()  
MsgBox "Now use the second electronic document and perform similar observations. This document  
will test for print size access when there is no contextual cueing. Consumer should use his  
her glasses.", vbOKOnly, "Instructions for Electronic Document 2"  
RefreshJaws  
tabComputer.Tab = 4  
cboPointSize(4).SetFocus  
End Sub
```

```
Private Sub cmdEDoc2_LostFocus()  
tabComputer.Tab = 4  
cboPointSize(4).SetFocus  
End Sub
```

```
Private Sub cmdEdoc2a_Click()  
MsgBox "(Use the computer's print size preference from this last test as your recommendation  
)" & vbNewLine & vbNewLine & _  
"RULE: If accuracy, speed, duration, and a working distance of no less than  
inches is maintained using a standard monitor with the standard OS settings, continue. If not  
introduce larger monitors, and print enhancing software.", vbOKOnly, "Electronic Document 2 Rule"  
tabComputer.Tab = 5  
cmdPrintEnhancing.SetFocus  
frmComputer.Caption = "Print Enhancing Software"  
End Sub
```

```
Private Sub cmdEdoc2a_LostFocus()  
If CheckForTab Then  
tabComputer.Tab = 5  
cmdPrintEnhancing.SetFocus  
frmComputer.Caption = "Print Enhancing Software"  
End If  
End Sub
```

```
Private Sub cmdEyeMed_Click()  
frmEyeData.Show vbModal  
RefreshJaws  
tabComputer.Tab = 0  
cmdRule.SetFocus  
End Sub
```

```
Private Sub RefreshJaws()  
tabComputer.Tab = 9  
cboYesNo(8).SetFocus  
End Sub
```

```
Private Sub cmdFactors_Click()  
MsgBox "Additional Factors that influence time required for training" & vbNewLine & vbNewLine  
"-3=consumer desires minimal proficiency, +3=consumer desires high proficiency", vbOKOnly  
"Additional Factors"  
RefreshJaws  
tabComputer.Tab = 11  
cboFactors(0).SetFocus  
End Sub
```

```
Private Sub cmdGoal_Click()  
MsgBox "Areas of recommendation instruction (Goals) -- Consumer Agreement Yes/No", vbOKOnly, "  
s/Objective Instructions"  
RefreshJaws  
tabComputer.Tab = 8  
chkRecommend(0).SetFocus  
End Sub
```

```
Private Sub cmdGoals_Click()
```

```
IsGoalsClicked = True
```

```
DisplayRecommendedServices
```

```
RefreshJaws
```

```
tabComputer.Tab = 12
```

```
txtAssessment(14).SetFocus
```

```
CountGoals
```

```
IsGoalsClicked = False
```

```
End Sub
```

```
Private Sub DisplayRecommendedServices()
```

```
Dim str2 As String
```

```
Dim x As Integer
```

```
x = 0
```

```
Str = "Based on your responses, the following services would be beneficial to you:" & vbNewLine
```

```
Goals that the consumer agreed to
```

```
If (cboYesNo(6) = "Yes") Then ' Goal
```

```
Str = Str & "Typing" & vbNewLine
```

```
Else
```

```
For x = 0 To 7
```

```
If (cboAgree(x) = "Yes") Then
```

```
Str = Str & "Typing" & vbNewLine
```

```
Exit For
```

```
End If
```

```
Next x
```

```
End If
```

```
If (cboYesNo(7) = "Yes") Then ' Goal
```

```
Str = Str & "Use Windows/Word Processor" & vbNewLine
```

```
Else
```

```
For x = 8 To 11 ' Objectives
```

```
If cboAgree(x) = "Yes" Then
```

```
Str = Str & "Use Windows/Word Processor" & vbNewLine
```

```
Exit For
```

```
End If
```

```
Next x
```

```
End If
```

```
If (cboYesNo(8) = "Yes") Then ' Goal
```

```
Str = Str & "Print Enlarging/Email" & vbNewLine
```

```
Else
```

```
For x = 12 To 14 ' Objectives
```

```
If cboAgree(x) = "Yes" Then
```

```
Str = Str & "Print Enlarging/Email" & vbNewLine
```

```
Exit For
```

```
End If
```

```
Next x
```

```
End If
```

```
If (cboYesNo(9) = "Yes") Then ' Goal
```

```
Str = Str & "Screen Reading/Email" & vbNewLine
```

```
Else
```

```
For x = 15 To 17 ' Objectives
```

```
If cboAgree(x) = "Yes" Then
```

```
Str = Str & "Screen Reading/Email" & vbNewLine
```

```
Exit For
```

```
End If
```

```
Next x
```

```
End If
```

```

If (cboYesNo(10) = "Yes") Then      ' Goal
    Str = Str & "Print Enhancing/Internet" & vbNewLine
Else                                ' Objectives
    For x = 18 To 22
        If cboAgree(x) = "Yes" Then
            Str = Str & "Print Enhancing/Internet" & vbNewLine
            Exit For
        End If
    Next x
End If

If (cboYesNo(11) = "Yes") Then      ' Goal
    Str = Str & "Screen Reading/Internet" & vbNewLine
Else                                ' Objectives
    For x = 23 To 27
        If cboAgree(x) = "Yes" Then
            Str = Str & "Screen Reading/Internet" & vbNewLine
            Exit For
        End If
    Next x
End If

If (cboYesNo(12) = "Yes") Then      ' Goal
    Str = Str & "Assistive Technology" & vbNewLine
Else                                ' Objectives
    For x = 28 To 35
        If cboAgree(x) = "Yes" Then
            Str = Str & "Assistive Technology" & vbNewLine
            Exit For
        End If
    Next x
End If

MsgBox Str, vbOKOnly, "Goals that the consumer agreed to"

End Sub

Private Sub CountGoals()
    Dim i As Integer

    For i = 0 To cboAgree.Count - 1
        Select Case i
            Case 0 To 7
                If cboAgree(i).Text = "Yes" Then chkGoals(0) = 1
            Case 8 To 11
                If cboAgree(i).Text = "Yes" Then chkGoals(1) = 1
            Case 12 To 14
                If cboAgree(i).Text = "Yes" Then chkGoals(2) = 1
            Case 15 To 17
                If cboAgree(i).Text = "Yes" Then chkGoals(3) = 1
            Case 18 To 22
                If cboAgree(i).Text = "Yes" Then chkGoals(4) = 1
            Case 23 To 27
                If cboAgree(i).Text = "Yes" Then chkGoals(5) = 1
            Case 28 To 35
                If cboAgree(i).Text = "Yes" Then chkGoals(6) = 1
        End Select
    Next i
End Sub

Private Sub cmdIdentify_Click()
    MsgBox "Number of times it took to correctly identify a word, and means, other than repeat  

    sed to allow the consumer to identify the word such as verbal cueing, reading the word letter,  

    letter, etc. (narrative)", vbOKOnly, "Question 4 Instruction"
    RefreshJaws
    tabComputer.Tab = 6

```

```
txtAssessment(12).SetFocus
End Sub
```

```
Private Sub cmdKeyboardInstruct_Click(Index As Integer)
MsgBox "Request consumer's preference for large print and/or speech to confirm keystrokes. Demstrate appropriate software until the consumer selects a font size and/or speech speed and tone. vbOKOnly, "Instructions for Keyboard Skills Testing"
RefreshJaws
tabComputer.Tab = 3.
cmdKeyboardRating.SetFocus
End Sub
```

```
Private Sub cmdKeyboardRating_Click()
MsgBox "if rating is 1 through 4 refer to Keyboarding (goal 1). Continue assessment". & vbNewLine & _
    "if 5 through 7 must schedule at least 20 hours in typing lab (2 hrs/day)" & _
    "if 8 through 10 skill sufficient", vbOKOnly, "Keyboard Skills Rating"
RefreshJaws
tabComputer.Tab = 3
cboLocateKeys.SetFocus
End Sub
```

```
Private Sub cmdKeyRating_Click()
MsgBox "(1 could not find the home row, 5 had moderate difficulty locating the home row, 10 had difficulty locating with correct hand positions)" & vbNewLine & _
    "If rating is 1 through 4 refer to Keyboarding (goal 1). Continue assessment" & vbNewLine & _
    "if rating is 5 through 7 must schedule at least 20 hours in typing lab (2 hrs/day)" & vbNewLine & _
    "if rating is 8 through 10 skill sufficient", vbOKOnly, "Key Scale and Identification Rating"
RefreshJaws
tabComputer.Tab = 3
cmdTyping.SetFocus
End Sub
```

```
Private Sub cmdLearning_Click()
MsgBox "Visual Learner - The use of residual vision for obstacle detection (dynamic and static) and to gather/verify environmental information (orientation) needed for safe movement/travel.", vbOKOnly, "Learning Strategy Types"
MsgBox "Visual/Tactual/Auditory Learner - The combination of the use of residual vision, hearing and a tactual mobility tool (e.g. prescription or support cane) for identification, obstacle detection (dynamic and static) and to gather/verify environmental information (orientation) needed for safe movement/travel.", vbOKOnly, "Learning Strategy Types"
MsgBox "Tactual/Auditory Learner - The utilization of tactual (e.g. protective techniques, mobility tool) and auditory methodologies for obstacle detection (dynamic and static) and in combination with communication skills to gather/verify environmental information (orientation) needed for safe movement/travel.", vbOKOnly, "Learning Strategy Types"
MsgBox "Visual/Tactual Learner - The utilization of visual, tactual and proprioceptive strategies for obstacle detection (dynamic and static) and in combination with appropriate communication methodologies to gather/verify environmental information (orientation) needed for safe movement/travel.", vbOKOnly, "Learning Strategy Types"
MsgBox "Tactual Learner/Traveler - The utilization of tactual and proprioceptive strategies for obstacle detection (dynamic and static) and in combination with appropriate communication methodologies to gather/verify environmental information (orientation) needed for safe movement/travel.", vbOKOnly, "Learning Strategy Types"
RefreshJaws
tabComputer.Tab = 11
cboStrategy.SetFocus
End Sub
```

```
Private Sub cmdNext_Click()
```

```
    If tabComputer.Tab = 12 Then
        MsgBox "You have reached the last page of the computer assessment!", vbInformation + vbOKOnly, "Next Page"
    Else
        tabComputer.Tab = tabComputer.Tab + 1
    End If
End Sub
```

```

frmComputer.Caption = tabComputer.Caption
If tabComputer.Tab = 1 Then lstOtherConditions.SetFocus
If tabComputer.Tab = 2 Then cboHighestEduc.SetFocus
If tabComputer.Tab = 3 Then cmdSpelling.SetFocus
If tabComputer.Tab = 4 Then cmdAccess.SetFocus
If tabComputer.Tab = 5 Then cmdPrintEnhancing.SetFocus
If tabComputer.Tab = 6 Then cmdSpeechInstruction.SetFocus
If tabComputer.Tab = 7 Then cmdNoteInstruct.SetFocus
If tabComputer.Tab = 8 Then cmdGoal.SetFocus
If tabComputer.Tab = 9 Then chkRecommend(2).SetFocus
If tabComputer.Tab = 10 Then chkRecommend(5).SetFocus
If tabComputer.Tab = 11 Then cmdLearning.SetFocus
If tabComputer.Tab = 12 Then txtAssessment(16).SetFocus
End If

End Sub

Private Sub cmdNExtRecord_Click()

With rstComputer
.MoveNext
If .AbsolutePosition = .RecordCount Then
    'Have record for this client and
    d it to form
    cmdNExtRecord.Enabled = False
    cmdPreviousRecord.Enabled = True
Else
    cmdPreviousRecord.Enabled = True
    cmdNExtRecord.Enabled = True
End If

txtCurrentRecord = .AbsolutePosition
txtSequenceNo = ![AssessmentSequence]
LoadDataToControls rstComputer, cboStaffs, False
LoadDataToControls rstComputer, txtAssessment, True
LoadDataToControls rstComputer, cboVisualFields, False
LoadDataToControls rstComputer, cboYesNo, True
LoadDataToControls rstComputer, lstDevices, False
LoadDataToControls rstComputer, lstOtherConditions, False
LoadDataToControls rstComputer, lstAffectTraining, False
LoadDataToControls rstComputer, cboUseComputer, False
LoadDataToControls rstComputer, cboOperatingSystem, False
LoadDataToControls rstComputer, lstTypeTraining, False
LoadDataToControls rstComputer, cboKeyboardSkill, False
LoadDataToControls rstComputer, cboRateYourTyping, False
LoadDataToControls rstComputer, cboSpelling, False
LoadDataToControls rstComputer, cboLocateKeys, False
LoadDataToControls rstComputer, cboTypingSpeed, False
LoadDataToControls rstComputer, cboFontSize, True
LoadDataToControls rstComputer, cboEnhancingName, False
LoadDataToControls rstComputer, cboMagnification, False
LoadDataToControls rstComputer, cboSpeech, False
LoadDataToControls rstComputer, cboModality, True
LoadDataToControls rstComputer, cboNotes, True
LoadDataToControls rstComputer, cboStrategy, False
LoadDataToControls rstComputer, cboAgree, True
LoadDataToControls rstComputer, cboEquipment, True
LoadDataToControls rstComputer, cboAgencyServices, True
LoadDataToControls rstComputer, cboExternalServices, True
LoadDataToControls rstComputer, cboServiceLevel, True
LoadDataToControls rstComputer, cboFactors, True

If ![FinishDate] = "12:00:00 PM" Then
    cmdSave.Enabled = True
Else
    cmdSave.Enabled = False
End If
End With

tabComputer.Tab = 0
txtCurrentRecord.SetFocus

```

End Sub

```
Private Sub cmdNote1_Click()
MsgBox "(1 = errors on every step; 10 = complete accuracy on every step)" & vbNewLine & "[if 3
less refer to Academic Skills for note taking]", vbOKOnly, "Note-taking accuracy rating"
RefreshJaws
tabComputer.Tab = 7
cboNotes(0).SetFocus
End Sub
```

```
Private Sub cmdNote2_Click()
MsgBox "(1 = no steps were complete; 10 = every step was complete)" & vbNewLine & "[if 3 or les
refer to Academic Skills for note taking]", vbOKOnly, "Note-taking completeness rating"
RefreshJaws
tabComputer.Tab = 7
cboNotes(2).SetFocus
End Sub
```

```
Private Sub cmdNote3_Click()
MsgBox "(1 = could not use notes at all; 10 = can use notes with ease)", vbOKOnly, "Note-taking
tilization"
RefreshJaws
tabComputer.Tab = 7
cboNotes(4).SetFocus
End Sub
```

```
Private Sub cmdNote4_Click()
MsgBox "1 = could not start process; 10 = can correctly perform tasks" & vbNewLine & vbNewLine
-
      "At 2nd test, ask consumer if they could think of any other way to have done this e
cise better, if possible. If they have an alternate method, perform the note taking test agai
sing that method." & vbNewLine & vbNewLine & _
      "Rule: If the higher of the two scores is under 4, recommend Academic Skills to lea
methods of note taking.", vbOKOnly, "Consumer's Rating"
tabComputer.Tab = 8
cmdGoal.SetFocus
frmComputer.Caption = "Goals 1"
End Sub
```

```
Private Sub cmdNote4_LostFocus()
If CheckForTab Then
tabComputer.Tab = 8
cmdGoal.SetFocus
frmComputer.Caption = "Goals 1"
End If
End Sub
```

```
Private Sub cmdNoteInstruct_Click()
MsgBox "Note-Taking - (This test may be administered twice if the consumer chooses)", vbOKOnly,
Note Taking Instruction"
RefreshJaws
tabComputer.Tab = 7
cboModality(0).SetFocus
End Sub
```

```
Private Sub cmdPrevious_Click()
If tabComputer.Tab = 0 Then
MsgBox "You are at the first page of the Computer Skills assessment!", vbInformation + vbI
ly, "Previous Page"
Else
tabComputer.Tab = tabComputer.Tab - 1
frmComputer.Caption = tabComputer.Caption
If tabComputer.Tab = 0 Then cboStaffs.SetFocus
If tabComputer.Tab = 1 Then 1stOtherConditions.SetFocus
If tabComputer.Tab = 2 Then cboHighestEduc.SetFocus
If tabComputer.Tab = 3 Then cmdSpelling.SetFocus
If tabComputer.Tab = 4 Then cmdAccess.SetFocus
If tabComputer.Tab = 5 Then cmdPrintEnhancing.SetFocus
If tabComputer.Tab = 6 Then cmdSpeechInstruction.SetFocus
```

```

    If tabComputer.Tab = 7 Then cmdNoteInstruct.SetFocus
    If tabComputer.Tab = 8 Then cmdGoal.SetFocus
    If tabComputer.Tab = 9 Then chkRecommend(2).SetFocus
    If tabComputer.Tab = 10 Then chkRecommend(5).SetFocus
    If tabComputer.Tab = 11 Then cmdLearning.SetFocus
End If
End Sub

Private Sub cmdPreviousRecord_Click()
    With rstComputer
        .MovePrevious
        If .AbsolutePosition = 1 Then
            'Have record for this client and load it
orm
            cmdPreviousRecord.Enabled = False
            cmdNextRecord.Enabled = True
        Else
            cmdPreviousRecord.Enabled = True
            cmdNextRecord.Enabled = True
        End If

        If ![FinishDate] = "12:00:00 PM" Then
            cmdSave.Enabled = True
        Else
            cmdSave.Enabled = False
        End If

        txtCurrentRecord = .AbsolutePosition
        txtSequenceNo = ![AssessmentSequence]
        LoadDataToControls rstComputer, cboStaffs, False
        LoadDataToControls rstComputer, txtAssessment, True
        LoadDataToControls rstComputer, cboVisualFields, False
        LoadDataToControls rstComputer, cboYesNo, True
        LoadDataToControls rstComputer, lstDevices, False
        LoadDataToControls rstComputer, lstOtherConditions, False
        LoadDataToControls rstComputer, lstAffectTraining, False
        LoadDataToControls rstComputer, cboUseComputer, False
        LoadDataToControls rstComputer, cboOperatingSystem, False
        LoadDataToControls rstComputer, lstTypeTraining, False
        LoadDataToControls rstComputer, cboKeyboardSkill, False
        LoadDataToControls rstComputer, cboRateYourTyping, False
        LoadDataToControls rstComputer, cboSpelling, False
        LoadDataToControls rstComputer, cboLocateKeys, False
        LoadDataToControls rstComputer, cboTypingSpeed, False
        LoadDataToControls rstComputer, cboPointSize, True
        LoadDataToControls rstComputer, cboEnhancingName, False
        LoadDataToControls rstComputer, cboMagnification, False
        LoadDataToControls rstComputer, cboSpeech, False
        LoadDataToControls rstComputer, cboModality, True
        LoadDataToControls rstComputer, cboNotes, True
        LoadDataToControls rstComputer, cboStrategy, False
        LoadDataToControls rstComputer, cboAgree, True
        LoadDataToControls rstComputer, cboEquipment, True
        LoadDataToControls rstComputer, cboAgencyServices, True
        LoadDataToControls rstComputer, cboExternalServices, True
        LoadDataToControls rstComputer, cboServiceLevel, True
        LoadDataToControls rstComputer, cboFactors, True
    End With

    tabComputer.Tab = 0
    txtCurrentRecord.SetFocus
End Sub

Private Sub cmdPrint_Click()
    Dim i As Integer

    If Printers.Count = 0 Then
        MsgBox "No printer found associating with this computer. Please add a printer to it and you can print out the Service Agreement!", vbInformation + vbOKOnly, "Printer"
    End If
End Sub

```

```

Else
    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select * From ComputerAgreement Where AssessmentSequence = " & txtSequenceNo, cnn,
    dOpenDynamic, adLockOptimistic
    If Not rst.EOF Then ' if there an existing report, delete it then create a new report
        rst.Delete
    End If
    rst.Close

    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select * From ComputerAgreement Where AssessmentSequence = " & txtSequenceNo, cnn,
    dOpenDynamic, adLockOptimistic
    With rst
        If .BOF And .EOF Then
            .AddNew
            ![CaseNumber] = txtCaseNumber
            ![Episode] = Episode
            ![AssessmentSequence] = txtSequenceNo
            ![ConsumerName] = txtConsumerName
            ![AssessmentDate] = txtAssessment(0)
            ![AssessorName] = cboStaffs.Text

            For i = 6 To 12 ' Goals
                If cboYesNo(i).ListIndex = -1 Then
                    .Fields(cboYesNo(i).Tag) = ""
                Else
                    .Fields(cboYesNo(i).Tag) = "Yes"
                End If
            Next

            For i = 0 To cboEquipment.Count - 1
                .Fields(cboEquipment(i).Tag) = cboEquipment(i).Text
            Next

            For i = 0 To cboAgencyServices.Count - 1
                .Fields(cboAgencyServices(i).Tag) = cboAgencyServices(i).Text
            Next

            For i = 0 To cboExternalServices.Count - 1
                .Fields(cboExternalServices(i).Tag) = cboExternalServices(i).Text
            Next

            .Fields("TimeFrame") = cboTimeFrame.Text
            .Fields("Comments") = txtAssessment(18).Text
            .Update
        End If
    End With
    Set ComputerAgreement.DataSource = rst
    ComputerAgreement.PrintReport
End If
End Sub

Private Sub cmdPrint1_Click()
    MsgBox "If the consumer can read the letters with no more than 2 errors at this size while at  

    orking distance of no less than 12 inches, continue to next phase. If more than 2 errors, or  

    a distance of less than 12 inches, or if print cannot be read at all, increase magnification  

    1 the consumer can read the print with less than 2 errors and at a distance of no less than 1  

    ches. ", vbOKOnly, "Print Enhancing Rule 1"
    RefreshJaws
    tabComputer.Tab = 5
    cmdPrint2.SetFocus
End Sub

Private Sub cmdPrint2_Click()
    MsgBox "If the consumer required print to be produced at a size of 4x or greater, or if the c  

    mer required more than 2 seconds to identify each letter, introduce speech as reinforcement o  

    r consideration as the primary access tool.", vbOKOnly, "Print Enhancing Rule 2"
    RefreshJaws
    tabComputer.Tab = 5
    cboEnhancingName.SetFocus

```



End Sub

```
Private Sub cmdPrintEnhancing_Click()
MsgBox "(On the operating system required by the consumer, load the program that will enhance a
enlarge the print displayed. Load the two documents provided with this test and begin testing
ith the size increased to 2x.)", vbOKOnly, "Print Enhancing Software"
RefreshJaws
tabComputer.Tab = 5
cmdPrint1.SetFocus
End Sub
```

```
Private Sub cmdRule_Click()
MsgBox "A) If acuity is greater than 20/400 demonstrate usefulness of speech as a reinforcement
o vision, to increase reading speed and to decrease fatigue" & vbNewLine & vbNewLine & _
"B) If 20/400 to 20/600 include speech output and/or Braille output in evaluation as a potentia
primary tool" & vbNewLine & vbNewLine & _
"C) If no vision or less than 20/600 introduce speech and/or Braille as primary access tool (wi
print enhancement as a potential reinforcement tool)", vbOKOnly, "Vision Rules"
RefreshJaws
tabComputer.Tab = 0
cboVisualFields.SetFocus
End Sub
```

```
Private Sub cmdRule2_Click()
MsgBox "if yes, prompt user: consider an access program that can be expanded to accommodate po:
ble further vision changes/loss", vbOKOnly, "Rule for Vision Loss"
RefreshJaws
tabComputer.Tab = 0
lstDevices.SetFocus
End Sub
```

```
Private Sub cmdRule3_Click()
MsgBox "If Personal use (not for work/school) skip to the 3 Keyboard Tests (Key Identificatio
Typing Speed, Combination/Sequencing).", vbOKOnly, "Rule for Learning Computer"
RefreshJaws
tabComputer.Tab = 1
cboOperatingSystem.SetFocus
End Sub
```

```
Private Sub cmdSave_Click()
```

```
CountScore
SaveRecords
```

End Sub

```
Private Sub SaveRecords()
```

```
Dim colComputer As New Collection
```

```
On Error GoTo errSave
```

```
colComputer.Add cboStaffs
colComputer.Add txtAssessment
colComputer.Add cboVisualFields
colComputer.Add cboYesNo
colComputer.Add lstDevices
colComputer.Add lstOtherConditions
colComputer.Add lstAffectTraining
colComputer.Add cboUseComputer
colComputer.Add cboOperatingSystem
colComputer.Add lstTypeTraining
colComputer.Add cboKeyboardSkill
colComputer.Add cboRateYourTyping
colComputer.Add cboSpelling
colComputer.Add cboLocateKeys
colComputer.Add cboTypingSpeed
colComputer.Add cboSequencing
colComputer.Add cboPointSize
colComputer.Add cboEnhancingName
```

```

colComputer.Add cboMagnification
colComputer.Add cboSpeech
colComputer.Add cboModality
colComputer.Add cboNoteSpeed1
colComputer.Add cboNoteSpeed2
colComputer.Add cboNoteAccuracy1
colComputer.Add cboNoteAccuracy2
colComputer.Add cboNotes
colComputer.Add cboStrategy
colComputer.Add cboFactors
colComputer.Add cboYesNo
colComputer.Add cboAgree
colComputer.Add cboEquipment
colComputer.Add cboAgencyServices
colComputer.Add cboExternalServices
colComputer.Add cboServiceLevel
colComputer.Add chkRecommend
colComputer.Add cboHighestEduc

```

```
Episode = txtEpisode
```

```
If SaveDataToDB(frmComputer, rstComputer, colComputer, "160", False) = True Then ' Save data to database
```

```
MsgBox "Successfully saved the Computer Assessment!", vbOKOnly, "Save"
```

```
If IsDate(txtAssessment(19)) Then ' Save the scores to the Interventions table
```

```

cnn.Execute ("Update Interventions Set PreConsumerScore = '" & txtConsumerScore & _
            "' , PreProviderScore = '" & txtProviderScore & "' Where ConsumerID = '" & ConsumerID & _
            "' And Agency = '" & AgencyCode & "' And Episode = '" & Episode & "' And CaseNumber = '" & CaseNumber & "' And InterventionID = '160' And (EndDate = #12:00:00 PM# or EndDate IS NULL)").

```

```

cnn.Execute ("UPDATE ComputerAssessment AS a INNER JOIN Interventions AS b ON (a.CaseNumber = b.CaseNumber) & _
            " AND (a.Episode = b.Episode) AND (a.ConsumerID = b.ConsumerID) SET a.InterventionSequence = b.InterventionSequence" & _
            " WHERE b.InterventionID = '160' AND b.CaseNumber = '" & CaseNumber & "' AND EndDate = #12:00:00PM# OR b.EndDate IS NULL)" & _
            " AND a.InterventionSequence = 0")

```

```
End If
```

```

frmComputer.Refresh
If IsNewRecord Then
    rstComputer.Requery
    txtSequenceNo = rstComputer![AssessmentSequence]
End If

```

```

frmComputer.Refresh
IsNewRecord = False
Set colComputer = Nothing
End If

```

```
Exit Sub
```

```

errSave:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub

```

```

Private Sub cmdSequencing_Click()
MsgBox "Provider Rating (1-10 Scale)" & vbNewLine & _
    "1 = had extreme difficulty performing sequencing task" & vbNewLine & _
    "5 = had moderate difficulty" & vbNewLine & _
    "10 = had no difficulties, no mistakes" & vbNewLine & _
    "Rule: If 4 or less, Recommend Keyboarding", vbOKOnly, "Sequencing Skills Scale"
RefreshJaws
tabComputer.Tab = 3

```

```
cboSequencing.SetFocus
End Sub
```

```
Private Sub cmdShortCut_Click()
    MsgBox "Press Alt P for Previous Page" & vbNewLine & "Press Alt N for Next Page"
End Sub
```

```
Private Sub cmdSpeech2_Click()
    MsgBox "1 = could not understand the speech at all" & vbNewLine & _
        "5 = had moderate difficulty" & vbNewLine & "10 = had no difficulty or errors understand:" & vbNewLine & _
        "g the speech", vbOKOnly, "Speech Access Scale"
    RefreshJaws
    tabComputer.Tab = 6
    cboSpeech.SetFocus
End Sub
```

```
Private Sub cmdSpeech3_Click()
    MsgBox "Have the consumer move through the sentences and say aloud the words that he/she heard. Assist them in repeating the words if he/she could not understand them the first time.", vbOKOnly, "Instructions for Speech"
    RefreshJaws
    tabComputer.Tab = 6
    cmdIdentify.SetFocus
End Sub
```

```
Private Sub cmdSpeechInstruction_Click()
    MsgBox "Administer this test to all speech and potential speech users. Load the Letter and Word Identification document provided for Speech Test One. " & _
        "Demonstrate how to use the right, left, down, and then the up arrows. Have the consumer use the arrow keys to move through letters to see if he/she can identify the letters accurately", vbOKOnly, "Instructions for Speech Access"
    MsgBox "As accuracy increases, have the consumer hold the control key down and hit the right & left arrows to hear one word at a time. Have the consumer move independently through two sentences moving letter by letter and word by word. As the consumer finishes the second sentence, demonstrate the command to have the screen reader spell out a word letter by letter and characters ad phonically.", vbOKOnly, "Instructions for Speech Access"
    RefreshJaws
    tabComputer.Tab = 6
    cmdSpeech2.SetFocus
End Sub
```

```
Private Sub cmdSpeechInstruction2_Click()
    MsgBox "As accuracy increases, have the consumer hold the control key down and hit the right & left arrows to hear one word at a time. Have the consumer move independently through two sentences moving letter by letter and word by word. As the consumer finishes the second sentence, demonstrate the command to have the screen reader spell out a word letter by letter and characters ad phonically.", vbOKOnly, "Instructions for Speech Access 2"
    RefreshJaws
    tabComputer.Tab = 6
    cmdSpeech2.SetFocus
End Sub
```

```
Private Sub cmdSpelling_Click()
    MsgBox "skip if consumer wants training for personal use" & vbNewLine & _
        "RULE: If CBVH sponsored, consumer must score a 4. If less, recommend Academic Skills - spelling", vbOKOnly, "Instructions for Spelling Skills"
    RefreshJaws
    tabComputer.Tab = 3
    cmdSpellingRating.SetFocus
End Sub
```

```
Private Sub cmdSpellingRanking_Click()
    MsgBox "Spelling Score (using the 1-10 Provider Scale)" & vbNewLine & _
        "0 through 10 correct answers Score = 1" & vbNewLine & _
        "11 through 21 correct answers Score = 4" & vbNewLine & _
        "22 through 34 correct answers Score = 7" & vbNewLine & _
        "35 through 45 correct answers Score = 10", vbOKOnly, "Spelling Skills Score"
    RefreshJaws
    tabComputer.Tab = 3
    txtAssessment(4).SetFocus
End Sub
```

End Sub

```
Private Sub cmdSpellingRating_Click()  
MsgBox "1 or 4 Extremely poor spelling. Must complete academic remediation before going on to te  
hnology training. Continue assessment" & vbNewLine & _  
"7 Eligible for all levels if the consumer enrolls in concurrent academic skills class to  
improve spelling." & vbNewLine & _  
"10 (for NYS Only) Has met mandatory spelling level for consumers sponsored by NYS CBVH :  
r ATC. If less than 4, CBVH will not refer for technology assessment or training (but continue :  
complete the 'Readiness Eval')", vbOKOnly, "Spelling Skills Rating"  
RefreshJaws  
tabComputer.Tab = 3  
cmdSpellingRanking.SetFocus  
End Sub
```

```
Private Sub cmdTestRule_Click()  
MsgBox "End assessment if consumer requested training for personal use", vbOKOnly, "Standardize  
Tests Rule"  
tabComputer.Tab = 4  
cmdAccess.SetFocus  
End Sub
```

```
Private Sub cmdTestRule_LostFocus()  
If CheckForTab Then  
frmComputer.Caption = "Access Mode"  
tabComputer.Tab = 4  
cmdAccess.SetFocus  
End If  
End Sub
```

```
Private Sub cmdTyping_Click()  
MsgBox "Instruct consumer to press Enter 3 times and rest 1 minute. Then commence typing speed  
st - have consumer type the same sentence non-stop for 3 minutes. Repeat twice." & vbNewLine &  
NewLine & "End assessment if consumer requested training for personal use", vbOKOnly, "Instruct  
ns for Typing Speed"  
RefreshJaws  
tabComputer.Tab = 3  
cmdTypingRating.SetFocus  
End Sub
```

```
Private Sub cmdTypingRating_Click()  
MsgBox "1 = Types 0 - 20 words per minute. Requires keyboarding." & vbNewLine & _  
"4 = Types 21 - 35 words per minute. Sufficient speed to enter any level but must practice in :  
s" & vbNewLine & _  
"7 = Types 36 - 50 words per minute. Must practice in labs to increase speed if goal is office  
rk." & vbNewLine & _  
"10 = Types 51+ words per minute. Sufficient speed, practice not required", vbOKOnly, "Typing :  
ed Rating"
```

End Sub

```
Private Sub Form_Load()
```

```
' Consumer Information  
txtConsumerName.Text = FirstName & " " & LastName  
txtPhoneNumber = PhoneNo  
txtCaseNumber = CaseNumber  
txtAddress = Address  
txtEpisode = Episode
```

```
' Load all control with codes and descriptions from the database  
LoadControlsItems cboStaffs, False  
LoadControlsItems cboVisualFields, False  
LoadControlsItems lstDevices, False  
LoadControlsItems lstOtherConditions, False  
LoadControlsItems lstAffectTraining, False  
LoadControlsItems cboUseComputer, False  
LoadControlsItems cboOperatingSystem, False  
LoadControlsItems lstTypeTraining, False  
LoadControlsItems cboKeyboardSkill, False
```

```

LoadControlsItems cboRateYourTyping, False
LoadControlsItems cboSpelling, False
LoadControlsItems cboPointSize, True
LoadControlsItems cboModality, True
LoadControlsItems cboEquipment, True
LoadControlsItems cboAgencyServices, True
LoadControlsItems cboExternalServices, True

```

```

Set rst = New ADODB.Recordset      ' Create new recordset for Consumer Agreement
rst.CursorLocation = adUseClient

```

```

Set rstComputer = New ADODB.Recordset      ' Create new recordset for Computer Assessments
rstComputer.CursorLocation = adUseClient

```

```

rstComputer.Open "Select * from ComputerAssessment Where ConsumerID = " & CustomerID & _
" And Agency = " & AgencyCode & " And Episode = " & txtEpisode & _
" And CaseNumber = " & CaseNumber & " And InterventionID = '160'" & _
" Order by AssessmentDate, FinishDate", cnn, adOpenKeyset, adLockOptimistic

```

```

With rstComputer

```

```

    txtRecordNo = .RecordCount

```

```

    If Not (.EOF And .BOF) Then      ' If there are assessments load the first one into the form

```

```

        .MoveFirst

```

```

        txtCurrentRecord = .AbsolutePosition

```

```

        txtSequenceNo = ![AssessmentSequence]

```

```

        LoadDataToControls rstComputer, cboStaffs, False

```

```

        LoadDataToControls rstComputer, txtAssessment, True

```

```

        LoadDataToControls rstComputer, cboVisualFields, False

```

```

        LoadDataToControls rstComputer, cboYesNo, True

```

```

        LoadDataToControls rstComputer, lstDevices, False

```

```

        LoadDataToControls rstComputer, lstOtherConditions, False

```

```

        LoadDataToControls rstComputer, lstAffectTraining, False

```

```

        LoadDataToControls rstComputer, cboUseComputer, False

```

```

        LoadDataToControls rstComputer, cboOperatingSystem, False

```

```

        LoadDataToControls rstComputer, lstTypeTraining, False

```

```

        LoadDataToControls rstComputer, cboKeyboardSkill, False

```

```

        LoadDataToControls rstComputer, cboRateYourTyping, False

```

```

        LoadDataToControls rstComputer, cboSpelling, False

```

```

        LoadDataToControls rstComputer, cboNoteSpeed1, False

```

```

        LoadDataToControls rstComputer, cboNoteSpeed2, False

```

```

        LoadDataToControls rstComputer, cboNoteAccuracy1, False

```

```

        LoadDataToControls rstComputer, cboNoteAccuracy2, False

```

```

        LoadDataToControls rstComputer, cboLocateKeys, False

```

```

        LoadDataToControls rstComputer, cboTypingSpeed, False

```

```

        LoadDataToControls rstComputer, cboSequencing, False

```

```

        LoadDataToControls rstComputer, cboPointSize, True

```

```

        LoadDataToControls rstComputer, cboEnhancingName, False

```

```

        LoadDataToControls rstComputer, cboMagnification, False

```

```

        LoadDataToControls rstComputer, cboSpeech, False

```

```

        LoadDataToControls rstComputer, cboModality, True

```

```

        LoadDataToControls rstComputer, cboNotes, True

```

```

        LoadDataToControls rstComputer, cboStrategy, False

```

```

        LoadDataToControls rstComputer, cboAgree, True

```

```

        LoadDataToControls rstComputer, cboEquipment, True

```

```

        LoadDataToControls rstComputer, cboAgencyServices, True

```

```

        LoadDataToControls rstComputer, cboExternalServices, True

```

```

        LoadDataToControls rstComputer, cboServiceLevel, True

```

```

        LoadDataToControls rstComputer, chkRecommend, True

```

```

        LoadDataToControls rstComputer, cboHighestEduc, False

```

```

        LoadDataToControls rstComputer, cboFactors, True

```

```

    If .RecordCount <= 1 Then

```

```

        cmdNextRecord.Enabled = False

```

```

        cmdPreviousRecord.Enabled = False

```

```

    ElseIf .RecordCount >= 2 Then

```

```

        cmdPreviousRecord.Enabled = False

```

```

        cmdNextRecord.Enabled = True

```

```

    End If

```

```

txtCurrentRecord = .AbsolutePosition

```

```
If ![FinishDate] <> "12:00:00 PM" And IsDate(![FinishDate]) Then
    cmdSave.Enabled = False
    cmdClear.Enabled = True
    cmdPrint.Enabled = True
Else
    cmdSave.Enabled = True
    cmdClear.Enabled = False
    cmdPrint.Enabled = False
End If

Else ' No ComputerAssessment records disable previous and next buttons
    cmdPreviousRecord.Enabled = False
    cmdNextRecord.Enabled = False
End If
End With

tabComputer.Tab = 0

End Sub

Private Sub Form_Unload(Cancel As Integer)
    If rstComputer.State = adStateOpen Then
        rstComputer.Close
    End If
    Set rstComputer = Nothing
End Sub

Private Sub lstDevices_LostFocus()
    If CheckForTab Then
        tabComputer.Tab = 1
        lstOtherConditions.SetFocus
        frmComputer.Caption = "Self Assessment 2"
    End If
End Sub

Private Sub lstOtherConditions_Click()
    If lstOtherConditions.ListIndex = 7 Then
        lstAffectTraining.Enabled = False
    Else
        lstAffectTraining.Enabled = True
    End If
End Sub

Private Sub lstOtherConditions_LostFocus()
    If CheckForTab = True Then
        If lstOtherConditions.ListIndex = 7 Then
            lstAffectTraining.Enabled = False
        Else
            lstAffectTraining.Enabled = True
        End If
    End If
End Sub

Private Sub tabComputer_Click(PreviousTab As Integer)
    If tabComputer.Tab = 0 Then
        frmComputer.Caption = "Self Assessment 1"
    ElseIf tabComputer.Tab = 1 Then
        frmComputer.Caption = "Self Assessment 2"
    ElseIf tabComputer.Tab = 2 Then
        frmComputer.Caption = "Self Assessment 3"
    ElseIf tabComputer.Tab = 3 Then
        frmComputer.Caption = "Standardized Test 1"
    ElseIf tabComputer.Tab = 4 Then
        frmComputer.Caption = "Access Mode"
    ElseIf tabComputer.Tab = 5 Then
        frmComputer.Caption = "Print Enhancing Software"
    ElseIf tabComputer.Tab = 6 Then
        frmComputer.Caption = "Speech Access"
    ElseIf tabComputer.Tab = 7 Then
```

```
frmComputer.Caption = "Note Taking"
ElseIf tabComputer.Tab = 8 Then
    frmComputer.Caption = "Goals 1"
ElseIf tabComputer.Tab = 9 Then
    frmComputer.Caption = "Goals 2"
ElseIf tabComputer.Tab = 10 Then
    frmComputer.Caption = "Goals 3"
ElseIf tabComputer.Tab = 11 Then
    frmComputer.Caption = "Learning Strategy and Factors"
ElseIf tabComputer.Tab = 12 Then
    frmComputer.Caption = "Consumer Agreement"
    CountGoals
    CountScore
End If
End Sub

Private Sub CountScore()
Dim i As Integer
Dim j As Integer
Dim tmpValue As Integer
Dim tmp1 As Integer
Dim tmp2 As Integer
Dim taskA As Long      ' Self-Assessment
Dim taskB As Long      ' Provider Rating
Dim taskC As Long      ' Standardized Tests
Dim taskD As Long      ' Learning Strategy
Dim taskE As Long      ' Factors
Dim OverallScore As Long

' Required fields to calculate a score

If cboStrategy.ListIndex = -1 Then
    MsgBox "You must select one learning strategy", vbOKOnly, "Learning Strategy"
    tabComputer.Tab = 11
    cboStrategy.SetFocus
Exit Sub
End If

For i = 0 To cboFactors.Count - 1
    If cboFactors(i).ListIndex = -1 Then
        MsgBox ("You must select a factor rating")
        tabComputer.Tab = 11
        cboFactors(i).SetFocus
        Exit Sub
    End If
Next

' Self-Assessment

taskA = 0

If cboKeyboardSkill.ListIndex = -1 Then      ' user didn't select anything default to 4
    taskA = 4
Else
    taskA = cboKeyboardSkill.ItemData(cboKeyboardSkill.ListIndex)
End If

If cboRateYourTyping.ListIndex = -1 Then      ' user didn't select anything default to 4
    tmpValue = 4
Else
    tmpValue = cboRateYourTyping.ItemData(cboRateYourTyping.ListIndex)
End If

taskA = taskA + tmpValue

If cboSpelling.ListIndex = -1 Then      ' user didn't select anything default to 4
    tmpValue = 4
Else
    tmpValue = cboSpelling.ItemData(cboSpelling.ListIndex)
```

End If

taskA = taskA + tmpValue

' Provider Rating

taskB = 0

If cboLocateKeys.ListIndex = -1 Then ' Key Identification  
taskB = 10

Else  
taskB = cboLocateKeys.ItemData(cboLocateKeys.ListIndex)

End If

If cboSequencing.ListIndex = -1 Then ' Combination Keys & Sequencing  
tmpValue = 10

Else  
tmpValue = cboSequencing.ItemData(cboSequencing.ListIndex)

End If

taskB = taskB + tmpValue

If cboSpeech.ListIndex = -1 Then ' Speech Rating  
tmpValue = 10

Else  
tmpValue = cboSpeech.ItemData(cboSpeech.ListIndex)

End If

taskB = taskB + tmpValue

' Note Taking

If cboNoteAccuracy1.ListIndex = -1 And cboNoteAccuracy2.ListIndex = -1 Then ' Note-taking /  
uracy

tmpValue = 10

Else

If cboNoteAccuracy1.ListIndex <> -1 And cboNoteAccuracy2.ListIndex <> -1 Then ' if there  
e 2 tests

tmp1 = cboNoteAccuracy1.ItemData(cboNoteAccuracy1.ListIndex) ' 1st Test

tmp2 = cboNoteAccuracy2.ItemData(cboNoteAccuracy2.ListIndex) ' 2nd Test

If tmp1 = tmp2 Then

tmpValue = tmp1

ElseIf tmp1 > tmp2 Then ' take the highest score from both t

s

tmpValue = tmp1

Else

tmpValue = tmp2

End If

End If

If cboNoteAccuracy1.ListIndex <> -1 And cboNoteAccuracy2.ListIndex = -1 Then ' 1st test

y

tmpValue = cboNoteAccuracy1.ItemData(cboNoteAccuracy1.ListIndex)

End If

End If

taskB = taskB + tmpValue

If cboNotes(0).ListIndex = -1 And cboNotes(1).ListIndex = -1 Then ' Note-taking completene  
tmpValue = 10

Else

If cboNotes(0).ListIndex <> -1 And cboNotes(1).ListIndex <> -1 Then ' if there were 2 te:  
tmp1 = cboNotes(0).ItemData(cboNotes(0).ListIndex) ' 1st Test

tmp2 = cboNotes(1).ItemData(cboNotes(1).ListIndex) ' 2nd Test

If tmp1 = tmp2 Then

tmpValue = tmp1



```

    ElseIf tmp1 > tmp2 Then
        ' take the highest score from both tes
    S
        tmpValue = tmp1
    Else
        tmpValue = tmp2
    End If
End If

If cboNotes(0).ListIndex <> -1 And cboNotes(1).ListIndex = -1 Then
    tmpValue = cboNotes(0).ItemData(cboNotes(0).ListIndex)
    ' 1st test only
End If

taskB = taskB + tmpValue

If cboNotes(2).ListIndex = -1 And cboNotes(3).ListIndex = -1 Then
    tmpValue = 10
    ' Note-taking Utilization
Else
    If cboNotes(2).ListIndex <> -1 And cboNotes(3).ListIndex <> -1 Then
        ' if there were 2 test
        tmp1 = cboNotes(2).ItemData(cboNotes(2).ListIndex)
        tmp2 = cboNotes(3).ItemData(cboNotes(3).ListIndex)
        ' 1st Test
        ' 2nd Test
        If tmp1 = tmp2 Then
            tmpValue = tmp1
        ElseIf tmp1 > tmp2 Then
            ' take the highest score from both tes
        S
            tmpValue = tmp1
        Else
            tmpValue = tmp2
        End If
    End If

    If cboNotes(2).ListIndex <> -1 And cboNotes(3).ListIndex = -1 Then
        tmpValue = cboNotes(2).ItemData(cboNotes(2).ListIndex)
        ' 1st test only
    End If
End If

taskB = taskB + tmpValue

If cboNotes(4).ListIndex = -1 And cboNotes(5).ListIndex = -1 Then
    tmpValue = 10
    ' Locating steps
Else
    If cboNotes(4).ListIndex <> -1 And cboNotes(5).ListIndex <> -1 Then
        ' if there were 2 tes
        tmp1 = cboNotes(4).ItemData(cboNotes(4).ListIndex)
        tmp2 = cboNotes(5).ItemData(cboNotes(5).ListIndex)
        ' 1st Test
        ' 2nd Test
        If tmp1 = tmp2 Then
            tmpValue = tmp1
        ElseIf tmp1 > tmp2 Then
            ' take the highest score from both t
        S
            tmpValue = tmp1
        Else
            tmpValue = tmp2
        End If
    End If

    If cboNotes(4).ListIndex <> -1 And cboNotes(5).ListIndex = -1 Then
        tmpValue = cboNotes(4).ItemData(cboNotes(4).ListIndex)
        ' 1st test only
    End If
End If

taskB = taskB + tmpValue

' Standardized Tests
taskC = 0

If Not IsNumeric(txtAssessment(4).Text) Then
    taskC = 10
    ' Spelling Test Score
Else

```

```
taskC = CLng(txtAssessment(4))
End If
```

```
If cboTypingSpeed.ListIndex = -1 Then      ' Typing Speed
    tmpValue = 10
Else
    tmpValue = cboTypingSpeed.ItemData(cboTypingSpeed.ListIndex)
End If
```

```
taskC = taskC + tmpValue
```

```
If cboNoteSpeed1.ListIndex = -1 And cboNoteSpeed2.ListIndex = -1 Then      ' Note-taking speed
    tmpValue = 10
Else
    If cboNoteSpeed1.ListIndex <> -1 And cboNoteSpeed2.ListIndex <> -1 Then      ' if there were 2
        sts
            tmp1 = cboNoteSpeed1.ItemData(cboNoteSpeed1.ListIndex)              ' 1st Test
            tmp2 = cboNoteSpeed2.ItemData(cboNoteSpeed2.ListIndex)              ' 2nd Test
            If tmp1 = tmp2 Then
                tmpValue = tmp1
            ElseIf tmp1 > tmp2 Then      ' take the highest score from both te
                tmpValue = tmp1
            Else
                tmpValue = tmp2
            End If
        End If
    End If
    If cboNoteSpeed1.ListIndex <> -1 And cboNoteSpeed2.ListIndex = -1 Then      ' 1st test only
        tmpValue = cboNoteSpeed1.ItemData(cboNoteSpeed1.ListIndex)
    End If
End If
```

```
taskC = taskC + tmpValue
```

```
' Learning Strategy
taskD = 0
```

```
taskD = cboStrategy.ItemData(cboStrategy.ListIndex)
```

```
' Factors
taskE = 0
tmpValue = 0
```

```
For j = 0 To cboFactors.Count - 1
    Select Case cboFactors(j).ItemData(cboFactors(j).ListIndex)
        Case -3: tmpValue = 120
        Case -2: tmpValue = 110
        Case -1: tmpValue = 105
        Case 0: tmpValue = 91
        Case 1: tmpValue = 80
        Case 2: tmpValue = 60
        Case 3: tmpValue = 25
    End Select
    taskE = taskE + tmpValue
Next
```

```
OverallScore = (((taskA * 8.3) + (taskB * 1.4) + Round((taskC * 10) / 3) + taskD + (taskE / 4)
5)
```

```
txtAssessment(16) = OverallScore
```

```
Select Case OverallScore
    Case 0 To 40:    txtAssessment(17) = "Level 5"
    Case 41 To 60:  txtAssessment(17) = "Level 4"
    Case 61 To 70:  txtAssessment(17) = "Level 3"
```

```

Case 71 To 80:   txtAssessment(17) = "Level 2"
Case 81 To 90:   txtAssessment(17) = "Level 1"
Case 91 To 100:  txtAssessment(17) = "No Service Needed"
End Select

txtAssessment(20) = CStr(taskA * 8.3)
txtAssessment(21) = CStr(taskB * 1.4)
txtAssessment(22) = CStr(Round((taskC * 10) / 3))
txtAssessment(23) = CStr(taskD)
txtAssessment(24) = CStr(taskE / 4)

txtConsumerScore = CStr(taskA * 8.3)
txtProviderScore = CStr(taskB * 1.4)

End Sub

Private Sub txtAssessment_LostFocus(Index As Integer)
    Dim a As Integer

    If Index = 0 Then
        If Not IsDate(Trim(txtAssessment(0))) Then
            MsgBox "Please enter a valid assessment date!", vbOKOnly, "Assessment Date"
            txtAssessment(0).SetFocus
        End If
    ElseIf Index = 9 Then
        If CheckForTab Then
            tabComputer.Tab = 6
            cmdSpeechInstruction.SetFocus
            frmComputer.Caption = "Speech Access"
        End If
    ElseIf Index = 14 Then
        If CheckForTab Then
            tabComputer.Tab = 7
            cmdNoteInstruct.SetFocus
            frmComputer.Caption = "Note Taking"
        End If
    ElseIf Index = 15 Then
        CountScore
        If CheckForTab Then
            tabComputer.Tab = 12
            txtAssessment(16).SetFocus
            frmComputer.Caption = "Consumer Agreement"
        End If
    ElseIf Index = 4 Then
        If Not IsNumeric(txtAssessment(4).Text) Then
            MsgBox "Please enter a number for the Spelling Score", vbOKOnly, "Spelling Score"
            tabComputer.Tab = 3
            txtAssessment(4).SetFocus
        Else
            If (Not CLng(txtAssessment(4)) = 1 And Not CLng(txtAssessment(4)) = 4 And Not CLng(txtAssessment(4)) = 7 And Not CLng(txtAssessment(4)) = 10) Then
                MsgBox "Please enter a 1 or 4 or 7 or 10 for the Spelling Score", vbOKOnly, "Spelling Score"
            End If
            tabComputer.Tab = 3
            txtAssessment(4).SetFocus
        End If
    End If

    End If

    If Index = 19 And CheckForTab Then
        If Trim(txtAssessment(19)) = "" And cmdSave.Enabled Then
            response = MsgBox("Did you finish the Computer Skills Assessment ?", vbQuestion + vbYesNo, "File Date")
            If response = vbYes Then
                MsgBox "Please enter the File Date !", vbInformation + vbOKOnly, "File Date"
                txtAssessment(19).SetFocus
            End If
        ElseIf IsDate(Trim(txtAssessment(19))) And cmdSave.Enabled Then
            response = MsgBox("Are you sure you finish the Computer Skills Assessment?", vbQuestion +

```

```
bYesNo, "File Date")
    If response = vbYes Then
        response = MsgBox("Would you like to save the Assessment information ?", vbQuestion + v
YesNo, "File Date")
        If response = vbYes Then
            SaveRecords
        End If
    Else
        txtAssessment(19) = ""
    End If
Else
    response = MsgBox("You reached the end of the Computer Skills Assessment", vbOKOnly, "").
    txtAssessment(19).SetFocus
End If
IsFileDateChecked = True
End If
End Sub
```

```

Option Explicit
Private rst As ADODB.Recordset      'resordset object variable
Private hasVision As String
Private IsArrowKey As Integer
Private IsShftTab As Boolean
Public IsNewRecord As Boolean
Private IsGoalsClicked As Boolean
Public IsFileDateChecked As Boolean
Private IsVisualInfoClicked As Boolean
Private rstEmployment As ADODB.Recordset
Dim LineCounter As Integer

Private Sub cboAgree_LostFocus(Index As Integer)
If Index = 8 Then
    tabEmployment.Tab = 10
    txtEmployment(61).SetFocus
End If
End Sub

Private Sub cboEmploymentStatus_Click(Index As Integer)

If IsArrowKey > 0 Then Exit Sub

' If employment status 1 is employed job in/not in jeopardy then
'   employment status 2 is either employed seeking/not seeking upgrade
' (unable to implement this rule, because other code relies answers based on status 2)

If Index = 1 Then
    If cboEmploymentStatus(0).ListIndex = 3 Then chkGoals(3).Value = 1
    If cboEmploymentStatus(0).ListIndex < 2 Then cboGeneralInfo(1).SetFocus ' Skip resume questi
                                         ' if employed in sta
End If
s 1

End Sub

Private Sub cboEmploymentStatus_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub

Private Sub cboEmploymentStatus_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If Index = 1 Then
    If KeyCode = vbKeyBack Then
        cboEmploymentStatus(Index).ListIndex = -1
    End If
End If
End Sub

Private Sub cboEmploymentStatus_LostFocus(Index As Integer)

If Index = 1 Then
    If cboEmploymentStatus(0).ListIndex = 3 Then chkGoals(3).Value = 1
    If cboEmploymentStatus(0).ListIndex < 2 Then cboGeneralInfo(1).SetFocus ' Skip resume quest:
                                         ' if employed in st:
End If
s 1

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboFactors_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

```

```
If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub
```

```
Private Sub cboFactors_Click(Index As Integer)
```

```
If IsArrowKey > 0 Then Exit Sub
```

```
If Index = 0 Then
    If cboFactors(0).ListIndex <> 6 Then
        cboFactors(1).SetFocus
    Else
        cboYesNo(10).SetFocus
    End If
ElseIf Index = 6 Then
    Call CountScore
End If
End Sub
```

```
Private Sub cboFactors_LostFocus(Index As Integer)
```

```
If Index = 0 Then
    If CheckForTab Then
        If cboFactors(0).ListIndex <> 6 Then cboFactors(1).SetFocus
    End If
ElseIf Index = 6 Then
    If CheckForTab Then Call CountScore
End If
```

```
If IsArrowKey > 0 Then IsArrowKey = 0
End Sub
```

```
Private Sub cboFactors1_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub
```

```
Private Sub cboFactors1_Click(Index As Integer)
```

```
If IsArrowKey > 0 Then Exit Sub
```

```
If Index = 4 Then Call CountScore1
End Sub
```

```
Private Sub cboFactors1_LostFocus(Index As Integer)
```

```
If Index = 4 Then
    If CheckForTab Then Call CountScore1
End If
```

```
If IsArrowKey > 0 Then IsArrowKey = 0
End Sub
```

```
Private Sub cboGeneralInfo_Click(Index As Integer)
```

```

If IsArrowKey > 0 Then Exit Sub

If Index = 0 Then
    If cboGeneralInfo(2).ListIndex = 2 Then
        chkGoals(14).Value = 1      ' trigger develop resume objective
        chkGoals(2).Value = 1      ' Acquire job seeking skills
        cboYesNo(20).ListIndex = 0
    End If
ElseIf Index = 2 Then
    If cboGeneralInfo(2).ListIndex <> 5 Then cboGeneralInfo(3).SetFocus
ElseIf Index = 8 Then
    If cboGeneralInfo(Index).ListIndex > -1 Then
        tabEmployment.Tab = 9
        CountGoals
        cboGeneralInfo(19).ListIndex = cboGeneralInfo(Index).ListIndex - 1
        txtEmployment(61) = cboGeneralInfo(19).Text
    End If
ElseIf Index = 9 Then
    ' What kind of work do you do?
    If cboGeneralInfo(9).ListIndex <> 3 Then cboGeneralInfo(10).SetFocus
    If cboGeneralInfo(9).ListIndex = 0 Or cboGeneralInfo(9).ListIndex = 1 Then
        chkGoals(0).Value = 1      ' Select a vocational goal
        chkGoals(4).Value = 1      ' Get a job
    End If
ElseIf Index = 10 Then
    If cboGeneralInfo(10).ListIndex <> 3 Then cboGeneralInfo(11).SetFocus
ElseIf Index = 11 Then
    If cboGeneralInfo(11).ListIndex <> 4 Then txtEmployment(21).SetFocus
ElseIf Index = 12 Then
    If IsShiftTab = True Then
        tabEmployment.Tab = 5
        txtEmployment(38).SetFocus
    End If
ElseIf Index = 13 Then
    If cboGeneralInfo(13).ListIndex = 0 Or cboGeneralInfo(13).ListIndex = 1 Then
        chkGoals(16).Value = 1
        chkGoals(2).Value = 1
    End If
ElseIf Index = 14 Then
    If cboGeneralInfo(14).ListIndex <> 2 Then cboJobDifficulty(16).SetFocus
ElseIf Index = 16 Then
    If cboGeneralInfo(17).ListIndex <> 5 Then cboFactors(1).SetFocus
ElseIf Index = 18 Then
    If cboGeneralInfo(Index).ListIndex > -1 Then
        tabEmployment.Tab = 9
        cmdGoals.SetFocus
        CountGoals
        cboGeneralInfo(19).ListIndex = cboGeneralInfo(Index).ListIndex - 1
        txtEmployment(61) = cboGeneralInfo(19).Text
    End If
End If
End Sub

```

```

Private Sub cboGeneralInfo_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

```

```

    If KeyCode = vbKeyUp Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyDown Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyRight Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyLeft Then
        IsArrowKey = KeyCode
    End If
End Sub

```

```

Private Sub cboGeneralInfo_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

```

```

    If KeyCode = vbKeyBack Then
        cboGeneralInfo(Index).ListIndex = -1
    End If
End Sub

```

End If  
End Sub

Private Sub cboGeneralInfo\_LostFocus(Index As Integer)

```
If Index = 2 Then
    If CheckForTab Then
        If cboGeneralInfo(2).ListIndex <> 5 Then cboGeneralInfo(3).SetFocus
    End If
ElseIf Index = 8 Then
    If CheckForTab Or cboGeneralInfo(Index).ListIndex > -1 Then
        tabEmployment.Tab = 9
        cmdGoals.SetFocus
        CountGoals
        cboGeneralInfo(19).ListIndex = cboGeneralInfo(Index).ListIndex - 1
        txtEmployment(61) = cboGeneralInfo(19).Text
    End If
ElseIf Index = 9 Then
    ' What kind of work do you do?
    If CheckForTab Then
        If cboGeneralInfo(9).ListIndex <> 3 Then cboGeneralInfo(10).SetFocus
        If cboGeneralInfo(9).ListIndex = 0 Or cboGeneralInfo(9).ListIndex = 1 Then
            chkGoals(8).Value = 1 ' Functional assessment inventory
            chkGoals(4).Value = 1 ' Get a job
        End If
    ElseIf CheckForShiftTab Then
        IsShiftTab = True
        tabEmployment.Tab = 0
        txtEmployment(8).SetFocus
    End If
ElseIf Index = 10 Then
    If CheckForTab Then
        If cboGeneralInfo(10).ListIndex <> 3 Then cboGeneralInfo(11).SetFocus
    End If
ElseIf Index = 11 Then
    If CheckForTab Then
        If cboGeneralInfo(11).ListIndex <> 4 Then txtEmployment(21).SetFocus
    End If
ElseIf Index = 12 Then
    If CheckForShiftTab Then
        IsShiftTab = True
        tabEmployment.Tab = 5
        txtEmployment(38).SetFocus
    End If
ElseIf Index = 13 Then
    If CheckForTab Then
        If cboGeneralInfo(13).ListIndex = 0 Or cboGeneralInfo(13).ListIndex = 1 Then
            chkGoals(16).Value = 1
            chkGoals(2).Value = 1
        End If
    End If
ElseIf Index = 14 Then
    If CheckForTab Then
        If cboGeneralInfo(14).ListIndex <> 2 Then cboJobDifficulty(16).SetFocus
    End If
ElseIf Index = 16 Then
    If CheckForTab Then
        If cboGeneralInfo(17).ListIndex <> 5 Then cboFactors(1).SetFocus
    End If
ElseIf Index = 18 Then
    If CheckForTab Or cboGeneralInfo(Index).ListIndex > -1 Then
        tabEmployment.Tab = 9
        CountGoals
        cboGeneralInfo(19).ListIndex = cboGeneralInfo(Index).ListIndex - 1
        txtEmployment(61) = cboGeneralInfo(19).Text
    End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub
```



```

Private Sub cboJobDifficulty_Click(Index As Integer)

If IsArrowKey > 0 Then Exit Sub

If Index = 1 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    cboScaleRating(18).SetFocus
  Else
    cboJobDifficulty(2).SetFocus
  End If
ElseIf Index = 2 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    cboYesNo(4).SetFocus
  Else
    cboJobDifficulty(3).SetFocus
  End If
ElseIf Index = 3 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    chkGoals(13).Value = 1      ' travel skills (o&m) - objective
    chkGoals(1).Value = 1      ' complete core skills - goal
    chkRecommendation(1).Value = 1 ' recommend o&m in dialog box
  End If
ElseIf Index = 4 Then
  If hasVision = "1" Then cboJobDifficulty(Index).ListIndex = 3
  If cboJobDifficulty(4).ListIndex = 0 Or cboJobDifficulty(4).ListIndex = 1 Then
    chkGoals(11).Value = 1      ' improve workplace lighting - objective
    chkGoals(1).Value = 1      ' complete core skills - goal
    chkRecommendation(0).Value = 1 ' recommend IL in dialog box
  End If
ElseIf Index = 5 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    chkGoals(12).Value = 1      ' organizational skills (IL) - objective
    chkGoals(1).Value = 1      ' complete core skills - goal
    chkRecommendation(0).Value = 1 ' recommend IL in dialog box
  End If
ElseIf Index = 6 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    chkRecommendation(0).Value = 1 ' recommend IL in dialog box
    If hasVision = "1" Then
      chkRecommendation(3).Value = 1 ' recommend LV in dialog box
    Else
      chkRecommendation(2).Value = 1 ' recommend computer skills in dialog box
    End If
  End If
ElseIf Index = 7 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    chkRecommendation(2).Value = 1
  End If
ElseIf Index = 8 Then
  tabEmployment.Tab = 3
ElseIf Index = 9 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    chkRecommendation(0).Value = 1
    cboScaleRating(0).SetFocus
  Else
    cboJobDifficulty(10).SetFocus
  End If
ElseIf Index = 10 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    chkRecommendation(0).Value = 1
    chkGoals(12).Value = 1      ' organizational skills
    chkGoals(1).Value = 1      ' complete core skills
  End If
ElseIf Index = 11 Then
  If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
    chkRecommendation(1).Value = 1
    chkGoals(13).Value = 1      ' travel skills (O&M)
    chkGoals(1).Value = 1      ' complete core skills
    cboScaleRating(2).SetFocus
  Else

```

```
cboJobDifficulty(11).SetFocus
End If
ElseIf Index = 12 Then
    If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
        cboScaleRating(3).SetFocus
    Else
        cboJobDifficulty(13).SetFocus
    End If
ElseIf Index = 13 Then
    If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
        cboScaleRating(7).SetFocus
    Else
        tabEmployment.Tab = 7
        cboJobDifficulty(14).SetFocus
    End If
ElseIf Index = 14 Then
    If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
        cboScaleRating(11).SetFocus
    Else
        cboJobDifficulty(15).SetFocus
    End If
ElseIf Index = 15 Then
    If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
        cboGeneralInfo(14).SetFocus
    Else
        cboJobDifficulty(16).SetFocus
    End If
End If
End Sub

Private Sub cboJobDifficulty_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyUp Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyDown Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyRight Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyLeft Then
        IsArrowKey = KeyCode
    End If
End Sub

Private Sub cboJobDifficulty_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboJobDifficulty(Index).ListIndex = -1
    End If
End Sub

Private Sub cboJobDifficulty_LostFocus(Index As Integer)
    If Index = 1 Then
        If CheckForTab Then
            If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
                cboScaleRating(18).SetFocus
            Else
                cboJobDifficulty(2).SetFocus
            End If
        End If
    ElseIf Index = 2 Then
        If CheckForTab Then
            If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
                cboYesNo(4).SetFocus
            Else
                cboJobDifficulty(3).SetFocus
            End If
        End If
    ElseIf Index = 3 Then
        If CheckForTab Then
```

```
    If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
        chkGoals(13).Value = 1
        chkGoals(1).Value = 1
        chkRecommendation(1).Value = 1
    End If
End If
ElseIf Index = 4 Then
    If CheckForTab Then
        If hasVision = "1" Then cboJobDifficulty(Index).ListIndex = -1
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            chkGoals(12).Value = 1
            chkGoals(1).Value = 1
            chkRecommendation(0).Value = 1
        End If
    End If
ElseIf Index = 5 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            chkGoals(12).Value = 1
            chkGoals(1).Value = 1
            chkRecommendation(0).Value = 1
        End If
    End If
ElseIf Index = 6 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            chkRecommendation(0).Value = 1 ' recommend IL in dialog box
            If hasVision = "1" Then ' consumer has no vision
                chkRecommendation(3).Value = 1 ' recommend LV in dialog box
            Else
                chkRecommendation(2).Value = 1 ' recommend computer skills in dialog box
            End If
        End If
    End If
ElseIf Index = 7 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            chkRecommendation(3).Value = 1
        End If
    End If
ElseIf Index = 8 Then
    If CheckForTab Then
        tabEmployment.Tab = 3
        cboRating(0).SetFocus
    End If
ElseIf Index = 9 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            chkRecommendation(0).Value = 1
            cboScaleRating(0).SetFocus
        Else
            cboJobDifficulty(10).SetFocus
        End If
    End If
ElseIf Index = 10 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            chkRecommendation(0).Value = 1
            chkGoals(12).Value = 1
            chkGoals(1).Value = 1 ' complete goal skills
        End If
    End If
ElseIf Index = 11 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            chkRecommendation(1).Value = 1
            chkGoals(13).Value = 1 ' travel skills (O&M)
            chkGoals(1).Value = 1 ' complete core skills
            cboScaleRating(2).SetFocus
        Else
```

```

        cboJobDifficulty(12).SetFocus
    End If
End If
ElseIf Index = 12 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            cboScaleRating(3).SetFocus
        Else
            cboJobDifficulty(13).SetFocus
        End If
    End If
ElseIf Index = 13 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            cboScaleRating(7).SetFocus
        Else
            tabEmployment.Tab = 7
            cboJobDifficulty(14).SetFocus
        End If
    End If
ElseIf Index = 14 Then
    If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
        cboScaleRating(11).SetFocus
    Else
        cboJobDifficulty(15).SetFocus
    End If
ElseIf Index = 15 Then
    If CheckForTab Then
        If cboJobDifficulty(Index).ListIndex = 0 Or cboJobDifficulty(Index).ListIndex = 1 Then
            cboGeneralInfo(14).SetFocus
        Else
            cboJobDifficulty(16).SetFocus
        End If
    End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboRating_Click(Index As Integer)

    If Index = 0 Then
        ' rate the consumer's presentation
        If cboRating(Index).ListIndex <= 4 Then
            chkGoals(10).Value = 1 ' grooming (IL) objective
            chkGoals(1).Value = 1 ' complete core skills
            chkRecommendation(0).Value = 1 ' recommend IL in dialog box
        End If
    ElseIf Index = 1 Then
        ' rate the consumer ability to move around the workplace
        If cboRating(Index).ListIndex <= 4 Then
            chkGoals(13).Value = 1 ' travel skills (o&m)
            chkGoals(1).Value = 1 ' complete core skills
            chkRecommendation(1).Value = 1 ' recommend mobility in dialog box
        End If
    ElseIf Index = 2 Then
        ' rate the adequacy of workplace lighting to meet consum
        s needs
        If cboRating(Index).ListIndex <= 4 Then
            chkRecommendation(1).Value = 1 ' Mobility
            chkGoals(11).Value = 1 ' Workplace Lighting
            chkGoals(1).Value = 1 ' complete core skills
        End If
    ElseIf Index = 3 Then
        ' rate the consumer's ability to locate items in his/her
        rkspace
        If cboRating(Index).ListIndex <= 4 Then
            chkRecommendation(0).Value = 1 ' IL
            chkGoals(12).Value = 1 ' organizational skills (IL)
            chkGoals(1).Value = 1 ' complete core sills
        End If
    ElseIf Index = 4 Then
        If cboRating(Index).ListIndex <= 4 Then chkRecommendation(2).Value = 1
    ElseIf Index = 5 Then

```

```

    If cboRating(Index).ListIndex <= 4 Then chkRecommendation(2).Value = 1
ElseIf Index >= 8 And Index <= 11 Then
    If cboRating(Index).ListIndex <= 4 Then
        chkGoals(2).Value = 1      ' Acquire job seeking skills
        chkGoals(15).Value = 1    ' Complete job interview training
    End If
ElseIf Index = 13 Then
    If cboRating(Index).ListIndex <= 4 Then
        chkRecommendation(0).Value = 1    ' IL recommendation
        chkGoals(1).Value = 1             ' Complete core skills
        chkGoals(10).Value = 1            ' Grooming (IL)
    End If
ElseIf Index = 15 Then
    If cboRating(Index).ListIndex <= 8 Then
        chkGoals(2).Value = 1      ' Acquire job seeking skills
        chkGoals(14).Value = 1    ' Develop a resume
    End If
End If
End Sub

Private Sub cboRating_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboRating(Index).ListIndex = -1
    End If
End Sub

Private Sub cboRating_LostFocus(Index As Integer)
    If Index = 0 Then
        ' rate the consumer's presentation
        If cboRating(Index).ListIndex <= 4 Then
            chkGoals(10).Value = 1    ' grooming (IL) objective
            chkGoals(1).Value = 1     ' complete core skills
            chkRecommendation(0).Value = 1 ' recommend IL in dialog box
        End If
    ElseIf Index = 1 Then
        ' rate the consumer ability to move around the workplace
        If Not CheckForShiftTab Then
            If cboRating(Index).ListIndex <= 4 Then
                chkGoals(13).Value = 1    ' travel skills (o&m)
                chkGoals(1).Value = 1     ' complete core skills
                chkRecommendation(1).Value = 1 ' recommend mobility in dialog box
            End If
        End If
    ElseIf Index = 2 Then
        ' rate the adequacy of workplace lighting to meet consumer needs
        If Not CheckForShiftTab Then
            If cboRating(Index).ListIndex <= 4 Then
                chkRecommendation(1).Value = 1    ' Mobility
                chkGoals(11).Value = 1             ' Workplace Lighting
                chkGoals(1).Value = 1              ' complete core skills
            End If
        End If
    ElseIf Index = 3 Then
        ' rate the consumer's ability to locate items in his/her workspace
        If Not CheckForShiftTab Then
            If cboRating(Index).ListIndex <= 4 Then
                chkRecommendation(0).Value = 1    ' IL
                chkGoals(12).Value = 1            ' organizational skills (IL)
                chkGoals(1).Value = 1            ' complete core skills
            End If
        End If
    ElseIf Index = 4 Then
        If Not CheckForShiftTab Then If cboRating(Index).ListIndex <= 4 Then chkRecommendation(2).Value = 1
    ElseIf Index = 5 Then
        If Not CheckForShiftTab Then If cboRating(Index).ListIndex <= 4 Then chkRecommendation(2).Value = 1
    ElseIf Index >= 8 And Index <= 11 Then
        If Not CheckForShiftTab Then

```

```

    If cboRating(Index).ListIndex <= 4 Then
        chkGoals(2).Value = 1      ' Acquire job seeking skills
        chkGoals(15).Value = 1    ' Complete job interview training
    End If
End If
ElseIf Index = 13 Then
    If Not CheckForShiftTab Then
        If cboRating(Index).ListIndex <= 4 Then
            chkRecommendation(0).Value = 1      ' IL recommendation
            chkGoals(1).Value = 1                ' Complete core skills
            chkGoals(10).Value = 1               ' Grooming (IL)
        End If
    End If
ElseIf Index = 15 Then
    If CheckForTab Then
        tabEmployment.Tab = 8
        cmdLearn2.SetFocus
        If cboRating(Index).ListIndex <= 8 Then
            chkGoals(2).Value = 1      ' Acquire job seeking skills
            chkGoals(14).Value = 1     ' Develop a resume
        End If
    End If
End If
End Sub

Private Sub cboScaleRating_Click(Index As Integer)

    If Index = 0 Then
        If cboScaleRating(0).ListIndex = 0 Or cboScaleRating(0).ListIndex = 1 Then
            chkGoals(10).Value = 1      ' recommend grooming (IL)
            chkGoals(1).Value = 1
            chkRecommendation(0).Value = 1
        End If
    ElseIf Index = 1 Then
        If cboScaleRating(1).ListIndex = 0 Or cboScaleRating(1).ListIndex = 1 Then
            chkGoals(10).Value = 1      ' recommend grooming (IL)
            chkGoals(1).Value = 1
        End If
    ElseIf Index = 10 Then
        tabEmployment.Tab = 7
    ElseIf Index = 18 Or Index = 17 Then
        If cboScaleRating(Index) = 0 Or cboScaleRating(Index) = 1 Then
            chkGoals(10).Value = 1      ' recommend grooming (IL)
            chkRecommendation(0).Value = 1      ' recommend IL in dialog box
        End If
    End If
End Sub

Private Sub cboScaleRating_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

    If KeyCode = vbKeyBack Then
        cboScaleRating(Index).ListIndex = -1
    End If
End Sub

Private Sub cboScaleRating_LostFocus(Index As Integer)

    If Index = 0 Or Index = 1 Then
        If CheckForTab Then
            If cboScaleRating(Index).ListIndex = 0 Or cboScaleRating(Index).ListIndex = 1 Then
                chkGoals(10).Value = 0      ' grooming (IL)
                chkGoals(1).Value = 1      ' complete goal skills
                ~ chkRecommendation(0).Value = 1
            End If
        End If
    ElseIf Index = 10 Then
        If CheckForTab Then tabEmployment.Tab = 7
    ElseIf Index = 18 Or Index = 17 Then
        If CheckForTab Then

```

```
If cboScaleRating(Index).ListIndex = 0 Or cboScaleRating(Index).ListIndex = 1 Then chkGo
(10).Value = 1
End If
End If
End Sub
```

```
Private Sub cboStaffs_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboStaffs.ListIndex = -1
End If
End Sub
```

```
Private Sub cboYesNo_Click(Index As Integer)
```

```
If IsArrowKey > 0 Then Exit Sub
```

```
If Index = 0 Then
    If cboYesNo(0).ListIndex <> 0 Then cboGeneralInfo(2).SetFocus
ElseIf Index = 3 Then
    If (cboYesNo(1).ListIndex = 0 Or cboYesNo(2).ListIndex = 0 Or _
        cboYesNo(3).ListIndex = 0) Then cboGeneralInfo(5).SetFocus
ElseIf Index = 5 Then
    If cboYesNo(Index).ListIndex = 1 Then
        chkGoals(13).Value = 1      ' recommend travel skills (O&M)
        chkRecommendation(1).Value = 1    ' recommend O&M in dialog box
    End If
ElseIf Index = 7 Then
    If cboYesNo(Index).ListIndex = 1 Then
        chkGoals(15).Value = 1      ' complete job interview training
        chkGoals(2).Value = 1      ' acquire job seeking skills
        cboJobDifficulty(10).SetFocus
    End If
End If
End Sub
```

```
Private Sub cboYesNo_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub
```

```
Private Sub cboYesNo_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboYesNo(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboYesNo_LostFocus(Index As Integer)
```

```
If Index = 0 Then
    If cboYesNo(0).ListIndex <> 0 And CheckForTab Then
        cboGeneralInfo(2).SetFocus
    End If
ElseIf Index = 3 Then
    If CheckForTab Then
        If (cboYesNo(1).ListIndex = 0 Or cboYesNo(2).ListIndex = 0 Or _
            cboYesNo(3).ListIndex = 0) Then cboGeneralInfo(5).SetFocus
    End If
ElseIf Index = 5 Then
    If cboYesNo(Index).Text = "No" Then
```

```

        chkGoals(10).Value = 1
        chkRecommendation(1).Value = 1
    End If
ElseIf Index = 7 Then
    If cboYesNo(Index).ListIndex = 1 Then
        chkGoals(15).Value = 1
        chkGoals(2).Value = 1
        Complete job interview training
        cboJobDifficulty(10).SetFocus
    End If
ElseIf Index = 25 Then
    If cboYesNo(Index).ListIndex = 1 Then cboJobDifficulty(6).ListIndex = 3
ElseIf Index = 26 Then
    If cboYesNo(Index).ListIndex = 1 Then cboJobDifficulty(7).ListIndex = 3
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cmdBenefitNote_Click()
MsgBox "If any entitlements are choses, discuss Impact of Employment on Entitlements with Bene
s", vbOKOnly, "Benefit Instructions"
End Sub

Private Sub cmdClear_Click()

IsNewRecord = True
clearData.txtEmployment, True
clearData.cboStaffs, False
clearData.cboGeneralInfo, True
clearData.chkGoals, True
clearData.cboYesNo, True
clearData.cboJobDifficulty, True
clearData.lstEmployment, True
clearData.cboFactors, True
clearData.cboFactors1, True
clearData.lstEmployment, True

If Not rstEmployment.EOF Then rstEmployment.MoveFirst
txtCurrentRecord = ""
cmdPrint.Enabled = False
cmdSave.Enabled = True
tabEmployment.Tab = 0
End Sub

Private Sub cmdClose_Click()
Unload Me
End Sub

Private Sub cmdEyeData_Click()
IsVisualInfoClicked = True
frmEyeData.Show vbModal
End Sub

Private Sub cmdInstruct_Click()
MsgBox "If the answer to any consumer self-rating question is a 1 or 2: recommend training
the consumer does not answer or indicates he/she is not interested in the task, mark as R re
d", vbOKOnly, "Self Assessment Instruction"
End Sub

Private Sub cmdLearn2_Click()
MsgBox "Visual Learner" & vbNewLine & "Visual/Tactual/Auditory Learner" & vbNewLine & _
"Tactual/Auditory Learner" & vbNewLine & "Visual/Tactual Learner" & vbNewLine & _
"Tactual Learner/Traveler", vbOKOnly, "Learning Strategy Types"
tabEmployment.Tab = 7
tabEmployment.Tab = 8
cboGeneralInfo(15).SetFocus
End Sub

Private Sub cmdLearning_Click()
MsgBox "Visual Learner" & vbNewLine & "Visual/Tactual/Auditory Learner" & vbNewLine & _

```



```

        "Tactual/Auditory Learner" & vbNewLine & "Visual/Tactual Learner" & vbNewLine & _
        "Tactual Learner/Traveler", vbOKOnly, "Learning Strategy Types"
tabEmployment.Tab = 2
tabEmployment.Tab = 3
cboGeneralInfo(6).SetFocus
End Sub

Private Sub cmdNext_Click()
If tabEmployment.Tab = 10 Then
    MsgBox "This is the last page of the Employment Services Assessment", vbInformation + vbOKC
    , "Next Page"
Else
    tabEmployment.Tab = tabEmployment.Tab + 1
    frmEmployment.Caption = getTabCaption(tabEmployment.Tab)
End If
End Sub

Private Sub cmdNextRecord_Click()

With rstEmployment
    .MoveNext
    If .AbsolutePosition = 1 Then
        cmdPreviousRecord.Enabled = False
        cmdNextRecord.Enabled = True
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = False
    End If

    txtCurrentRecord = .AbsolutePosition
    LoadDataToControls rstEmployment, cboEmploymentStatus, True
    LoadDataToControls rstEmployment, txtEmployment, True
    LoadDataToControls rstEmployment, cboStaffs, False
    LoadDataToControls rstEmployment, cboGeneralInfo, True
    LoadDataToControls rstEmployment, chkGoals, True
    LoadDataToControls rstEmployment, cboYesNo, True
    LoadDataToControls rstEmployment, cboJobDifficulty, True
    LoadDataToControls rstEmployment, lstEmployment, True
    LoadDataToControls rstEmployment, cboRating, True
    LoadDataToControls rstEmployment, cboAgree, True
    LoadDataToControls rstEmployment, cboFactors, True
    LoadDataToControls rstEmployment, cboFactors1, True

    If ![FinishDate] = "12:00:00 PM" Then
        cmdSave.Enabled = True
    Else
        cmdSave.Enabled = False
    End If
End With

tabEmployment.Tab = 0
txtCurrentRecord.SetFocus
End Sub

Private Sub cmdPrevious_Click()
If tabEmployment.Tab = 0 Then
    MsgBox "It is first page for Self Assessment already!", vbInformation + vbOKOnly, "Previous Page"
Else
    tabEmployment.Tab = tabEmployment.Tab - 1
    frmEmployment.Caption = getTabCaption(tabEmployment.Tab)
End If
End Sub

Private Sub cmdPreviousRecord_Click()

With rstEmployment

```

```

.MovePrevious
If .AbsolutePosition = 1 Then
    cmdPreviousRecord.Enabled = False
    cmdNextRecord.Enabled = True
ElseIf .AbsolutePosition < .RecordCount Then
    cmdPreviousRecord.Enabled = True
    cmdNextRecord.Enabled = True
ElseIf .RecordCount = .AbsolutePosition Then
    cmdPreviousRecord.Enabled = True
    cmdNextRecord.Enabled = False
End If

If ![FinishDate] = "12:00:00 PM" Then
    cmdSave.Enabled = True
Else
    cmdSave.Enabled = False
End If

txtCurrentRecord = .AbsolutePosition
LoadDataToControls rstEmployment, cboEmploymentStatus, True
LoadDataToControls rstEmployment, txtEmployment, True
LoadDataToControls rstEmployment, cboStaffs, False
LoadDataToControls rstEmployment, cboGeneralInfo, True
LoadDataToControls rstEmployment, chkGoals, True
LoadDataToControls rstEmployment, cboYesNo, True
LoadDataToControls rstEmployment, cboJobDifficulty, True
LoadDataToControls rstEmployment, lstEmployment, True
LoadDataToControls rstEmployment, cboRating, True
LoadDataToControls rstEmployment, cboAgree, True
LoadDataToControls rstEmployment, cboFactors, True
LoadDataToControls rstEmployment, cboFactors1, True

End With

tabEmployment.Tab = 0
txtCurrentRecord.SetFocus
End Sub

Private Sub cmdPrint_Click()
Dim i As Integer

If Printers.Count = 0 Then
    MsgBox "No printer found associating with this computer. Please add a printer to it and then you can print out the Service agreement! ", _
        vbInformation + vbOKOnly, "Printer"
Else
    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select * from employmentAgreement Where AssessmentSequence = " & rstEmployment![InterventionSequence], cnn, adOpenKeyset, adLockOptimistic
    If Not rst.EOF Then ' if there an existing report, delete it then create a new report
        rst.Delete
    End If
    rst.Close
    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select * from employmentAgreement Where AssessmentSequence = " & rstEmployment![InterventionSequence], cnn, adOpenKeyset, adLockOptimistic
    With rst
        If .BOF And .EOF Then
            .AddNew
            ![CaseNumber] = txtCaseNumber
            ![Episode] = txtEpisode
            ![AssessmentSequence] = rstEmployment![InterventionSequence]
            ![ConsumerName] = txtConsumerName
            ![AssessmentDate] = txtEmployment(0)
            ![Interviewer] = cboStaffs.Text
        For i = 0 To 7 - 1
            If chkGoals(i).Value = 0 Then
                .Fields(chkGoals(i).Tag) = ""
            Else
                .Fields(chkGoals(i).Tag) = "Yes"
            End If
        Next i
    End With
End If

```

```

        End If
    Next

    For i = 11 To 17
        .Fields(cboYesNo(i).Tag) = cboYesNo(i).Text
    Next

    For i = 20 To 25
        .Fields(cboGeneralInfo(i).Tag) = cboGeneralInfo(i).Text
    Next

    .Update
End If
End With
Set EmploymentAgreement.DataSource = rst
EmploymentAgreement.PrintReport
End If
End Sub

Private Sub cmdPrintFinal_Click()
Dim rsReport As New ADODB.Recordset
Dim rsReport2 As New ADODB.Recordset
Dim Filler As Integer
Dim Temp As Integer
Dim Counselor As String
Dim CaseNo As String
Dim PayerSequence As Integer
Dim PayerLocation As Integer
Dim AuthID As String
Dim Location As String
Dim Achieved As String
Dim NotAchieved As String
Dim Staff As String
Dim Visits As Integer
Dim ServiceHrs As Single
Dim ProgressSequence As Integer
Dim Period1 As String
Dim Period2 As String
Dim GoalDesc As String
Dim ObjDesc As String
Dim GAchieve As String
Dim OAchieve As String
Dim Ach As String
Dim NotAch As String
Dim GSeq As Integer
Dim Summary As String * 65000
Dim SumText As String
Dim BegL As Integer
Dim BlankPosition As Integer
Dim LearningProcess As String
Dim Techniques As String
Dim Responsibility As String
Dim Realistic As String
Dim TempStr As String
Dim CodeValue As String

Achieved = "X"
NotAchieved = "___"

Set rsReport = New Recordset
rsReport.CursorLocation = adUseClient

' Retrieve CBVH counselor, Case No and PayerSequence
rsReport.Open "Select PayerContact, IDWithPayer, PayerSequence, PayerLocation From PayerInfo
-
    " Where ConsumerId = " & CustomerID & " And Agency = " & AgencyCode & " " & _
    " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & " " & _
    " And PayerID = '150'", cnn, adOpenForwardOnly, adLockOptimistic

With rsReport
    If Not (.BOF Or .EOF) Then

```

```

        .MoveFirst
        Counselor = ![PayerContact]
        CaseNo = ![IDWithPayer]
        PayerSequence = ![PayerSequence]
        PayerLocation = ![PayerLocation]
    End If
End With
rsReport.Close

' Get the CBVH Location
Set rsReport = New Recordset
rsReport.CursorLocation = adUseClient
rsReport.Open "Select Description From PayerLocation" & _
    " Where PayerID = '150'" & _
    " And Code = '" & PayerLocation & "'", cnn, adOpenForwardOnly, adLockOptimistic

With rsReport
    If Not (.BOF Or .EOF) Then
        .MoveFirst
        Location = ![Description]
    End If
End With
rsReport.Close

' Get the Authorization Number for this Intervention
Set rsReport = New Recordset
rsReport.CursorLocation = adUseClient
rsReport.Open "Select AuthorizationID From PayerAuth PA, Interventions I" & _
    " Where PA.ConsumerID = '" & CustomerID & "' And PA.Agency = '" & AgencyCode & "' & _
    " And PA.Episode = '" & Episode & "' And PA.CaseNumber = '" & CaseNumber & "' & _
    " And PA.PayerSequence = '" & PayerSequence & "' & _
    " And PA.ConsumerID = I.ConsumerID And PA.Agency = I.Agency" & _
    " And PA.CaseNumber = I.CaseNumber And PA.Episode = I.Episode" & _
    " And PA.InterventionSequence = I.InterventionSequence" & _
    " And I.InterventionID = '050'", cnn, adOpenForwardOnly, adLockOptimistic

With rsReport
    If Not (.BOF Or .EOF) Then
        .MoveFirst
        AuthID = ![AuthorizationID]
    End If
End With
rsReport.Close

Set rsReport = Nothing

' Total Service Hours
Set rsReport = New Recordset
rsReport.CursorLocation = adUseClient

Visits = 0
ServiceHrs = 0

rsReport.Open "Select Duration From Visits V" & _
    " Where V.ConsumerID = '" & CustomerID & "' And V.Agency = '" & AgencyCode & "' & _
    " And V.Episode = '" & Episode & "' And V.CaseNumber = '" & CaseNumber & "' & _
    " And V.InterventionSequence = '" & rstEmployment![InterventionSequence], cnn, a
nForwardOnly, adLockOptimistic

With rsReport
    If Not (.BOF Or .EOF) Then
        .MoveFirst
        While .EOF = False
            Visits = Visits + ![Duration]
            .MoveNext
        Wend
    End If
End With
rsReport.Close

```

ServiceHrs = Visits / 60

Set rsReport = Nothing

' Service Period

Set rsReport = New Recordset

rsReport.CursorLocation = adUseClient

rsReport.Open "Select min(ServiceDate) as P1, max(ServiceDate) as P2 From Visits" & \_  
" Where ConsumerId = " & CustomerID & " And Agency = '" & AgencyCode & "'" & \_  
" And Episode = " & Episode & " And CaseNumber = '" & CaseNumber & "'" & \_  
" And InterventionSequence = " & rstEmployment![InterventionSequence], cnn, adOp  
rwardOnly, adLockOptimistic

With rsReport

If Not (.BOF Or .EOF) Then

.MoveFirst

Period1 = ![P1]

Period2 = ![P2]

End If

End With

rsReport.Close

Set rsReport = Nothing

' Get the Staff Assigned

Set rsReport = New Recordset

rsReport.CursorLocation = adUseClient

rsReport.Open "Select StaffName From Staffs, EmploymentAssessment E" & \_  
" Where E.ConsumerId = " & CustomerID & " And E.Agency = '" & AgencyCode & "'" & \_  
" And E.Episode = " & Episode & " And E.CaseNumber = '" & CaseNumber & "'" & \_  
" And E.AssessmentSequence = " & rstEmployment![AssessmentSequence] & \_  
" And Staffs.Id = E.Interviewer", cnn, adOpenForwardOnly, adLockOptimistic

With rsReport

If Not (.BOF Or .EOF) Then

.MoveFirst

Staff = ![StaffName]

End If

End With

rsReport.Close

Set rsReport = Nothing

' Retrieve Summary and Progress Sequence

Set rsReport = New Recordset

rsReport.CursorLocation = adUseClient

rsReport.Open "Select Summary, ProgressSeq From ProgressReport P" & \_  
" Where P.ConsumerId = " & CustomerID & " And P.Agency = '" & AgencyCode & "'" & \_  
" And P.Episode = " & Episode & " And P.CaseNumber = '" & CaseNumber & "'" & \_  
" And P.AssessmentSequence = " & rstEmployment![AssessmentSequence] & \_  
" And InterventionID = '050'", cnn, adOpenForwardOnly, adLockOptimistic

With rsReport

If Not (.BOF Or .EOF) Then

.MoveFirst

ProgressSequence = ![ProgressSeq]

Summary = ![Summary]

End If

End With

rsReport.Close

Set rsReport = Nothing

```

' End of Service Provider Report
Set rsReport = New Recordset
rsReport.CursorLocation = adUseClient

rsReport.Open "Select LearningProcess, Techniques, Responsibility, Realistic From Intervention
, Extent E" & _
    " Where I.ConsumerId = " & CustomerID & " And I.Agency = '" & AgencyCode & "'" & _
    " And I.Episode = " & Episode & " And I.CaseNumber = '" & CaseNumber & "'" & _
    " And I.InterventionSequence = " & rstEmployment![InterventionSequence] & _
    " And InterventionID = '050'", cnn, adOpenForwardOnly, adLockOptimistic

With rsReport
    If Not (.BOF Or .EOF) Then ,
        .MoveFirst
        LearningProcess = ![LearningProcess]
        Techniques = ![Techniques]
        Responsibility = ![Responsibility]
        Realistic = ![Realistic]
    End If
End With
rsReport.Close

Set rsReport = Nothing

Temp = 1

While Temp < 5
    If Temp = 1 Then CodeValue = LearningProcess
    If Temp = 2 Then CodeValue = Techniques
    If Temp = 3 Then CodeValue = Responsibility
    If Temp = 4 Then CodeValue = Realistic

    Set rsReport = New Recordset
    rsReport.CursorLocation = adUseClient

    rsReport.Open "Select Description From Extent" & _
        " Where Code = '" & CodeValue & "'", cnn, adOpenForwardOnly, adLockOptimistic

    With rsReport
        If Not (.BOF Or .EOF) Then
            .MoveFirst
            While .EOF = False
                TempStr = ![Description]
                .MoveNext
            Wend
        End If
    End With

    If Temp = 1 Then LearningProcess = TempStr
    If Temp = 2 Then Techniques = TempStr
    If Temp = 3 Then Responsibility = TempStr
    If Temp = 4 Then Realistic = TempStr

    rsReport.Close
    Temp = Temp + 1
Wend

Set rsReport = Nothing

PrintBlankLines (4)
Printer.FontSize = 12
Printer.Font = "Arial"
Printer.Print Spc(49); "Final Report"
Printer.Font.Bold = True
Printer.Print Spc(37); "Placement Services - Initial"
PrintBlankLines (2)

```

```

Printer.Print Spc(89); "58"
Printer.Print Spc(74); "Authorization No.:"
Printer.Font.Bold = False
Printer.Print Spc(82); AuthID
PrintBlankLines (2)
Temp = Len(txtConsumerName)
Filler = 52 - Temp
Printer.Print vbTab & "Consumer Name: " & txtConsumerName; Spc(Filler); "Case No: " & CaseNo
Printer.Print
Printer.Print vbTab & "CBVH Counselor:"; Spc(1); Counselor
Printer.Print vbTab & "CBVH D.O.:"; Spc(6); Location
PrintBlankLines (3)
Printer.Font.Bold = True
Printer.Print vbTab & vbTab & vbTab & "Final Report Service Outcome"
Printer.Font.Bold = False
Printer.Print
Printer.Print vbTab & vbTab & vbTab & vbTab & "___" & Achieved & "___ Outcome Achieved"
Printer.Print vbTab & vbTab & vbTab & vbTab & "___" & NotAchieved & "___ Outcome Not Achieved"
Printer.Print
Printer.Print
Printer.Font.Bold = True
Printer.Print vbTab & vbTab & vbTab & "Total Hour Units Rendered:" & vbTab;
Printer.FontBold = False
Printer.Print ServiceHrs
Printer.Print
Printer.Font.Bold = True
Printer.Print vbTab & vbTab & vbTab & "Total Service Period: " & vbTab & vbTab;
Printer.Font.Bold = False
Printer.Print Period1 & " to " & Period2
PrintBlankLines (4)
Printer.Font.Underline = True
Printer.Print vbTab & "Lighthouse" & vbTab & vbTab & vbTab & Staff & vbTab & vbTab & Date
Printer.Font.Underline = False
Printer.Print vbTab & "Agency Name" & vbTab & vbTab & vbTab & "Agency Signature" & vbTab & vbTab & "Date"
PrintBlankLines (3)
Printer.Print vbTab & "___ Report Accepted" & vbTab & vbTab; "___" & vbTab & vbTab &
"/___/___"
Printer.Print vbTab & "___ Not Accepted" & vbTab & vbTab & "CBVH Signature" & vbTab & vbTab &
te"

' Get the Goals from the IRA
Printer.NewPage
PrintBlankLines (5)
Printer.Print Spc(68); "Achieved"; Spc(5); "Not Achieved"
Ach = " "
NotAch = " "
LineCounter = 0

Set rsReport = New Recordset
rsReport.CursorLocation = adUseClient

rsReport.Open "Select GA.GoalID, Achievement, Description, GoalSeq From GoalAchieve GA, Goals
& -
" Where GA.ConsumerId = " & CustomerID & " And GA.Agency = " & AgencyCode & "
-
" And GA.Episode = " & Episode & " And GA.CaseNumber = " & CaseNumber & " &
" And GA.ProgressSeq = " & ProgressSequence & " And GA.GoalID = G.GoalID" &
" And GA.InterventionID = '050'", cnn, adOpenForwardOnly, adLockOptimistic

With rsReport
If Not (.BOF Or .EOF) Then
.MoveFirst
While .EOF = False
GoalDesc = Mid(![Description], 1, 45)
GAchieve = ![Achievement]
GSeq = ![GoalSeq]
If GAchieve = "1" Or GAchieve = "2" Or GAchieve = "3" Or GAchieve = "4" Or GAchieve
" Then
NotAch = "x"

```

```

        Ach = " "
    Else
        NotAch = " "
        Ach = "X"
    End If
    Filler = 52 - Len(GoalDesc)

    Printer.Print vbTab & "Goal: " & GoalDesc; Spc(Filler); vbTab & Ach & vbTab & vbTab
otAch
    LineCounter = LineCounter + 1
    CheckLineCounter (LineCounter)

    ' Get the objectives associated with this goal
    Set rsReport2 = New Recordset
    rsReport2.CursorLocation = adUseClient

    rsReport2.Open "Select OA.GoalID, Achievement, Description From ObjectiveAchieve OA,
ectives O" & _
        " Where OA.ConsumerID = " & CustomerID & " And OA.Agency = " & AgencyCode & "
        " And OA.Episode = " & Episode & " And OA.CaseNumber = " & CaseNumber & " & _
        " And OA.GoalSeq = " & GSeq & " And OA.GoalID = O.GoalID And OA.ObjectiveID = O.
ectiveID" & _
        " And OA.Achievement IS NOT NULL", cnn, adOpenForwardOnly, adLockOptimistic

    With rsReport2
        If Not (.BOF Or .EOF) Then
            .MoveFirst
            While .EOF = False
                ObjDesc = Mid(![Description], 1, 36)
                OAchieve = ![Achievement]
                If OAchieve = "1" Or OAchieve = "2" Or OAchieve = "3" Or OAchieve = "4" O:
chieve = "5" Then
                    NotAch = "X"
                    Ach = " "
                Else
                    NotAch = " "
                    Ach = "X"
                End If
                Temp = Len(ObjDesc)
                Filler = 41 - Temp
                Printer.Print Spc(13); "Objective: " & ObjDesc; Spc(Filler); vbTab & Ach
bTab & vbTab & NotAch
                    LineCounter = LineCounter + 1
                    CheckLineCounter (LineCounter)
                    .MoveNext
            Wend
        End If
    End With
    rsReport2.Close
    Set rsReport2 = Nothing ' End of Objectives loop

    PrintBlankLines (1)
    LineCounter = LineCounter + 1
    CheckLineCounter (LineCounter)
    .MoveNext
    Wend
End If
End With
rsReport.Close
Set rsReport = Nothing ' End of Goals Loop

' Summary Narrative
PrintBlankLines (1).
BegL = 1
Printer.Print vbTab & "Narrative:"
LineCounter = LineCounter + 2
CheckLineCounter (LineCounter)

```



```

SumText = Mid(Summary, BegL, 90)
BlankPosition = InStrRev(SumText, " ")
SumText = Mid(SumText, 1, BlankPosition)
Printer.Print vbTab & SumText
BegL = BlankPosition + 1
LineCounter = LineCounter + 1
CheckLineCounter (LineCounter)

```

```

For Temp = 1 To 20
    SumText = Mid(Summary, BegL, 90)
    BlankPosition = InStrRev(SumText, " ")
    SumText = Mid(SumText, 1, BlankPosition)
    Printer.Print vbTab & SumText
    BegL = BlankPosition + BegL
    LineCounter = LineCounter + 1
    CheckLineCounter (LineCounter)
Next

```

```

PrintBlankLines (2)
LineCounter = LineCounter + 2
CheckLineCounter (LineCounter)

```

```

' End of Service Provider Report
Printer.Print vbTab & "Post-Service Provider Questionnaire"
Printer.Print
LineCounter = LineCounter + 2
CheckLineCounter (LineCounter)
Printer.Print vbTab & " 1) Extent consumer took part in learning process." & vbTab & vbTab & LearningProcess
Printer.Print vbTab & " 2) Extent consumer used techniques s/he learned." & vbTab & vbTab & Techniques
Printer.Print vbTab & " 3) Extent consumer took responsibility for change." & vbTab & vbTab & Responsibility
Printer.Print vbTab & " 4) Extent consumer was realistic about expectations." & vbTab & vbTab & Realistic

```

```
Printer.EndDoc
```

```
End Sub
```

```
Private Sub PrintBlankLines(num As Integer)
```

```

Do While (num <> 0)
    Printer.Print
    num = num - 1
Loop

```

```
End Sub
```

```
Private Sub CheckLineCounter(num As Integer)
```

```

If num = 60 Then
    Printer.NewPage
    LineCounter = 0
End If

```

```
End Sub
```

```

Private Sub cmdQuestion5_Click()
    MsgBox "(if answer to question 5 is No, terminate assessment or recommend external benefits)",
    OKOnly, "Question 5 Instruction"
End Sub

```

```

Private Sub cmdRecommendation_Click()
    Dim tmpStr As String
    Dim i As Integer

```

```

tmpStr = "Based on Consumer Self-Assessment And Provider Ratings,"
If chkRecommendation.Count = 0 Then
    tmpStr = tmpStr & " no additional services are recommended."

```

```
ElseIf chkRecommendation.Count = 1 Then
    tmpStr = tmpStr & " the following additional service is recommended."
Else
    tmpStr = tmpStr & " the following additional services are recommended."
End If

tmpStr = tmpStr & vbNewLine & vbTab
For i = 0 To chkRecommendation.Count - 1
    If chkRecommendation(i).Value = 1 Then tmpStr = tmpStr & chkRecommendation(i).Tag & vbNewLine
    & vbTab
Next

MsgBox tmpStr, vbOKOnly, "Recommended Services"
tabEmployment.Tab = 2
tabEmployment.Tab = 3
txtEmployment(45).SetFocus
End Sub

Private Sub cmdRecommendation_LostFocus()

If CheckForShiftTab Then
    If cboEmploymentStatus(0).ListIndex = 2 Or cboEmploymentStatus(0).ListIndex = 3 Then
        IsShiftTab = True
        tabEmployment.Tab = 8
        cboFactors(6).SetFocus
    Else
        IsShiftTab = True
        tabEmployment.Tab = 3
        cboFactors1(4).SetFocus
    End If
End If
End Sub

Private Sub cmdRecommendation1_Click()
Dim tmpStr As String
Dim i As Integer

tmpStr = "Based on Consumer Self-Assessment And Provider Ratings,"
If chkRecommendation.Count = 0 Then
    tmpStr = tmpStr & " no additional services are recommended."
ElseIf chkRecommendation.Count = 1 Then
    tmpStr = tmpStr & " the following additional service is recommended."
Else
    tmpStr = tmpStr & " the following additional services are recommended."
End If

tmpStr = tmpStr & vbNewLine & vbTab
For i = 0 To chkRecommendation.Count - 1
    If chkRecommendation(i).Value = 1 Then tmpStr = tmpStr & chkRecommendation(i).Tag & vbNewLine
    & vbTab
Next

MsgBox tmpStr, vbOKOnly, "Recommended Services"
tabEmployment.Tab = 7
tabEmployment.Tab = 8
txtEmployment(47).SetFocus

End Sub

Private Sub cmdSave_Click()
Dim colEmployment As New Collection
Dim i As Integer

On Error GoTo errSave

colEmployment.Add cboEmploymentStatus
colEmployment.Add txtEmployment
colEmployment.Add cboStaffs
colEmployment.Add cboGeneralInfo
colEmployment.Add chkGoals
```

```
colEmployment.Add cboYesNo
colEmployment.Add cboJobDifficulty
colEmployment.Add lstEmployment
colEmployment.Add cboFactors
colEmployment.Add cboFactors1
colEmployment.Add cboRating
colEmployment.Add cboAgree
```

```
If IsDate(txtEmployment(50)) Then ' If filing the assessment make sure the learning strategy
                                ' and the factors are there
```

```
    If cboEmploymentStatus(0).ListIndex = 2 Then ' Job Placement
        If cboGeneralInfo(15).ListIndex = -1 Then
            tabEmployment.Tab = 8
            cboGeneralInfo(15).SetFocus
            MsgBox "You have to select one item from the Learning Strategy!", vbInformation + vbOKOnly
y, "Strategy Learning"
            Exit Sub
        End If
```

```
        For i = 0 To cboFactors.Count - 1
            If cboFactors(i).ListIndex = -1 Then
                tabEmployment.Tab = 8
                cboFactors(i).SetFocus
                MsgBox "You have to select one of the items from Factors!", vbInformation + vbOKOnly
"Factors"
                Exit Sub
            End If
        Next
```

```
    ElseIf cboEmploymentStatus(0).ListIndex = 0 Or cboEmploymentStatus(0).ListIndex = 1 Then ' Jo
Retention
        If cboGeneralInfo(6).ListIndex = -1 Then
            tabEmployment.Tab = 3
            cboGeneralInfo(6).SetFocus
            MsgBox "You have to select one item from the Learning Strategy!", vbInformation + vbOKOnly
y, "Strategy Learning"
            Exit Sub
        End If
```

```
        For i = 0 To cboFactors1.Count - 1
            If cboFactors1(i).ListIndex = -1 Then
                tabEmployment.Tab = 3
                cboFactors1(i).SetFocus
                MsgBox "You have to select one of the items from Factors!", vbInformation + vbOKOnly
"Factors"
                Exit Sub
            End If
        Next
    End If
End If
```

```
If SaveDataToDB(frmEmployment, rstEmployment, colEmployment, "050", False) Then 'Call
    blic function to save data to DB
    MsgBox "Successfully saved the Assessment information!", vbOKOnly, "Save"
```

```
If IsDate(txtEmployment(50)) Then
```

```
    cnn.Execute "Update Interventions Set PreConsumerScore = '" & txtConsumerScore & "', Pre
viderScore = '" & _
        txtProviderScore & "' Where ConsumerID = '" & CustomerID & "' And Agency = '" & AgencyC
& _
        "' And Episode = '" & Episode & "' And CaseNumber = '" & CaseNumber & _
        "' And InterventionID = '050' And (EndDate = #12:00:00 PM# or EndDate IS NULL)"

    cnn.Execute "UPDATE EmploymentAssessment AS a INNER JOIN Interventions AS b ON (a.CaseNum
= b.CaseNumber)" & _
        " AND (a.Episode = b.Episode) AND (a.ConsumerID = b.ConsumerID) SET a.InterventionSeq
ce = b.InterventionSequence" & _
        ". WHERE b.InterventionID = '050' and b.CaseNumber = '" & CaseNumber & "' AND (b.EndDate
```

```

= #12:00:00PM# OR b.EndDate IS NULL)" & _
    " AND a.InterventionSequence = 0"
End If

If IsNewRecord Then
    rstEmployment.Requery
    txtSequencyNo = rstEmployment![AssessmentSequence]
End If

IsNewRecord = False
Set colEmployment = Nothing
End If
Exit Sub

errSave:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub

Private Sub Form_Activate()

On Error GoTo Errhandler

If IsVisualInfoClicked Then Exit Sub
txtConsumerName = FirstName & " " & LastName
txtPhone = PhoneNo
txtAddress = Address
txtCaseNumber = CaseNumber
txtEpisode = Episode

IsArrowKey = 20                                'Avoid Combo click event
LoadControlsItems cboStaffs, False              'Add every description from database to combobox on
tbox

Set rst = New ADODB.Recordset                    'create recordset object
rst.CursorLocation = adUseClient
rst.Open "Select AgeAtOnset, SelfVisualAny, Employed, JobJeopardy From Triage Where ConsumerID = " & _
    CustomerID, cnn, adOpenForwardOnly, adLockReadOnly
With rst
    If Not (.BOF Or .EOF) Then
        .MoveFirst
        hasVision = ![SelfVisualAny]

        If ![employed] = 1 Then
            If ![jobjeopardy] = 1 Then
                cboEmploymentStatus(0).ListIndex = 0
            Else
                cboEmploymentStatus(0).ListIndex = 1
            End If
        ElseIf ![employed] = 2 Then
            cboEmploymentStatus(0).ListIndex = 2
        End If
    End If
End With
txtIntervention = "Employment"
rst.Close

Set rstEmployment = New ADODB.Recordset
rstEmployment.CursorLocation = adUseClient
rstEmployment.Open "Select * From Employmentassessment Where ConsumerID = " & CustomerID & _
    " And Agency = " & AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & _
    CaseNumber & " And InterventionID = '050' Order By AssessmentSequence DESC", cnn, _
    adOpenKeyset, adLockOptimistic
With rstEmployment
    txtRecordNo = .RecordCount
    If Not (.EOF And .BOF) Then                    'Have record for this client and load it to
m
        txtCurrentRecord = .AbsolutePosition
        LoadDataToControls rstEmployment, cboEmploymentStatus, True
        LoadDataToControls rstEmployment, txtEmployment, True
    End If
End With

```

```

LoadDataToControls rstEmployment, cboStaffs, False
LoadDataToControls rstEmployment, cboGeneralInfo, True
LoadDataToControls rstEmployment, chkGoals, True
LoadDataToControls rstEmployment, cboYesNo, True
LoadDataToControls rstEmployment, cboJobDifficulty, True
LoadDataToControls rstEmployment, lstEmployment, True
LoadDataToControls rstEmployment, cboRating, True
LoadDataToControls rstEmployment, cboAgree, True
LoadDataToControls rstEmployment, cboFactors, True
LoadDataToControls rstEmployment, cboFactors1, True

If .RecordCount > .AbsolutePosition Then cmdNextRecord.Enabled = True

If ![FinishDate] <> "12:00:00 PM" And IsDate(![FinishDate]) Then
    cmdSave.Enabled = False
    cmdPrint.Enabled = True
    cmdClear.Enabled = True
Else
    cmdSave.Enabled = True
    cmdPrint.Enabled = False
    cmdClear.Enabled = False
End If

Else
    cmdNextRecord.Enabled = False
    cmdPreviousRecord.Enabled = False
    txtCurrentRecord = 0
End If
End With

tabEmployment.Tab = 0
txtConsumerName.SetFocus
IsArrowKey = 0
Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"
End Sub

Private Sub Form_Unload(Cancel As Integer)
Dim response As Integer
Dim i As Integer

If IsFileDateChecked Then
    If rst.State = adStateOpen Then rst.Close
    Set rst = Nothing
    rstEmployment.Close
    Set rstEmployment = Nothing
    Set frmEmployment = Nothing
Else
    If Trim(txtEmployment(40)) = "" And cmdSave.Enabled Then
        response = MsgBox("Have you finished the Employment Assessment ?", vbQuestion + vbYesNo,
            "File Date")
        If response = vbYes Then
            MsgBox "Please enter the File Date !", vbInformation + vbOKOnly, "File Date"
            txtEmployment(40).SetFocus
            Cancel = True
        Else
            If rst.State = adStateOpen Then rst.Close
            Set rst = Nothing
            rstEmployment.Close
            Set rstEmployment = Nothing
            Set frmEmployment = Nothing
        End If
    ElseIf IsDate(Trim(txtEmployment(40))) And cmdSave.Enabled Then
        response = MsgBox("Are you sure you finished with the Employment Assessment?", vbQuestion + vbYesNo, "File Date")
        If response = vbYes Then
            response = MsgBox("Please use Save button to save the assessment information", vbQuestion + vbOKOnly, "File Date")

```

```

    If response = vbYes Then
        cmdSave.SetFocus
        Cancel = True
    End If
    ElseIf cmdSave.Enabled = True Then
        txtEmployment(40) = ""
    End If
    ElseIf IsDate(Trim(txtEmployment(40))) And Not cmdSave.Enabled Then
        If rst.State = adStateOpen Then rst.Close
        Set rst = Nothing
        rstEmployment.Close
        Set rstEmployment = Nothing
        Set frmEmployment = Nothing
    End If
End If
End Sub

Private Function getTabCaption(tabIndex As Integer) As String

Select Case tabIndex
    Case 0
        getTabCaption = "A. General Information"
    Case 1
        getTabCaption = "B. Job Retention - Employment"
    Case 2
        getTabCaption = "C. Job Retention - Self Assessment"
    Case 3
        getTabCaption = "D. Job Retention - Service Provider Rating, Factors And Learning Strategies"
    Case 4
        getTabCaption = "E. Counselling, Placement & Supported - Vocational"
    Case 5
        getTabCaption = "F. Counselling, Placement & Supported - Job History"
    Case 6
        getTabCaption = "G. Counselling, Placement & Supported - Consumer Self-Assessment"
    Case 7
        getTabCaption = "H. Counselling, Placement & Supported - Consumer Self-Assessment And Service Provider Rating"
    Case 8
        getTabCaption = "I. Counselling, Placement & Supported - Service Provider Rating, Learning Strategy And Factors"
    Case 9
        getTabCaption = "J. Goals"
    Case 10
        getTabCaption = "K. Employment Services Agreement - Service Agreement"
End Select

End Function

Private Sub CountScore()
    Dim i As Integer
    Dim tmpValue As Integer
    Dim taskA As Long
    Dim taskB As Long
    Dim taskD As Long
    Dim taskE As Long
    Dim avgScore As Single

    If cboGeneralInfo(15).ListIndex = -1 Then
        tabEmployment.Tab = 8
        cboGeneralInfo(15).SetFocus
        MsgBox "You had to select one of items from Strategy Learning!", vbInformation + vbOKOnly, "Strategy Learning"
        Exit Sub
    End If

    For i = 0 To cboFactors.Count - 1
        If cboFactors(i).ListIndex = -1 Then
            tabEmployment.Tab = 8
            cboFactors(i).SetFocus
            MsgBox "You had to select one of items from Factors!", vbInformation + vbOKOnly, "Factors"
        End If
    Next i

```

```

Exit Sub
End If
Next

For i = 9 To cboJobDifficulty.Count - 1 'count the score for task A
    If cboJobDifficulty(i).ListIndex = -1 Then
        taskA = taskA + 4
    Else
        tmpValue = cboJobDifficulty(i).ItemData(cboJobDifficulty(i).ListIndex)
        If tmpValue = 5 Then 'User select "Refused", count 1 for this item
            taskA = taskA + 1
        Else
            taskA = taskA + tmpValue
        End If
    End If
Next

If cboGeneralInfo(14).ListIndex = -1 Then
    taskA = taskA + 4
Else
    tmpValue = cboGeneralInfo(14).ItemData(cboGeneralInfo(14).ListIndex)
    If tmpValue = 5 Then 'User select "Refused", count 1 for this item
        taskA = taskA + 1
    Else
        taskA = taskA + tmpValue
    End If
End If

tmpValue = 0
For i = 6 To cboRating.Count - 1
    If cboRating(i).ListIndex = -1 Then 'user does'nt select one, count 10 for this item
        taskB = taskB + 10
    Else
        tmpValue = cboRating(i).ItemData(cboRating(i).ListIndex)
        If tmpValue = 11 Then 'User select "Refused", count 1 for this item
            taskB = taskB + 10
        Else
            taskB = taskB + tmpValue
        End If
    End If
Next

taskD = cboGeneralInfo(15).ItemData(cboGeneralInfo(15).ListIndex)

tmpValue = 0
For i = 0 To cboFactors.Count - 1
    Select Case cboFactors(i).ItemData(cboFactors(i).ListIndex)
        Case -3: tmpValue = 120
        Case -2: tmpValue = 110
        Case -1: tmpValue = 105
        Case 0: tmpValue = 91
        Case 1: tmpValue = 80
        Case 2: tmpValue = 60
        Case 3: tmpValue = 25
    End Select
    taskE = taskE + tmpValue
Next

txtConsumerScore = taskB * 1.9
txtProviderScore = taskA * 1#
avgScore = ((taskA * 1.9) + (taskB * 1#) + taskD + (taskE / 7)) / 4
txtEmployment(47) = CLng(avgScore)
txtEmployment(51) = Format(taskA * 1.9, "###")
txtEmployment(52) = Format(taskB * 1#, "###")
txtEmployment(53) = taskD
txtEmployment(54) = Format(taskE / 7, "###")
txtEmployment(55) = Format(avgScore, "###")

Select Case CLng(avgScore)

```

```

Case 0 To 40: txtEmployment(48) = "Level 4"
Case 41 To 60: txtEmployment(48) = "Level 3"
Case 61 To 80: txtEmployment(48) = "Level 2"
Case 81 To 90: txtEmployment(48) = "Level 1"
Case 91 To 100: txtEmployment(48) = "No Service Needed"
End Select
End Sub

Private Sub CountScore1()
Dim i As Integer
Dim tmpValue As Integer
Dim taskA As Long
Dim taskB As Long
Dim taskD As Long
Dim taskE As Long
Dim avgScore As Single

If cboGeneralInfo(6).ListIndex = -1 Then
    tabEmployment.Tab = 3
    cboGeneralInfo(6).SetFocus
    MsgBox "You had to select one of items from Strategy Learning!", vbInformation + vbOKOnly, "Strategy Learning".
    Exit Sub
End If

For i = 0 To 4
    If cboFactors1(i).ListIndex = -1 Then
        tabEmployment.Tab = 3
        cboFactors1(i).SetFocus
        MsgBox "You had to select one of items from Factors!", vbInformation + vbOKOnly, "Factors"
        Exit Sub
    End If
Next

For i = 0 To 8
    'count the score for task A
    If cboJobDifficulty(i).ListIndex = -1 Then
        taskA = taskA + 4
    Else
        tmpValue = cboJobDifficulty(i).ItemData(cboJobDifficulty(i).ListIndex)
        If tmpValue = 5 Then
            'User select "Refused", count 1 for this item
            taskA = taskA + 1
        Else
            taskA = taskA + tmpValue
        End If
    End If
Next

tmpValue = 0
For i = 0 To 5
    If cboRating(i).ListIndex = -1 Then
        'user does'nt select one, count 10 for this item
        taskB = taskB + 10
    Else
        tmpValue = cboRating(i).ItemData(cboRating(i).ListIndex)
        If tmpValue = 11 Then
            'User select "Refused", count 1 for this item
            taskB = taskB + 10
        Else
            taskB = taskB + tmpValue
        End If
    End If
Next

taskD = cboGeneralInfo(6).ItemData(cboGeneralInfo(6).ListIndex)

tmpValue = 0
For i = 0 To cboFactors1.Count - 1
    Select Case cboFactors1(i).ItemData(cboFactors1(i).ListIndex)
        Case -3: tmpValue = 120
        Case -2: tmpValue = 110
        Case -1: tmpValue = 105
        Case 0: tmpValue = 91
    End Select

```



```

Case 1: tmpValue = 80
Case 2: tmpValue = 60
Case 3: tmpValue = 25

```

```
End Select
```

```
taskE = taskE + tmpValue
```

```
Next
```

```
txtConsumerScore = taskB * 1.7
```

```
txtProviderScore = taskA * 2.8
```

```
avgScore = ((taskA * 2.8) + (taskB * 1.7) + taskD + (taskE / 5)) / 5
```

```
txtEmployment(45) = CLng(avgScore)
```

```
txtEmployment(56) = Format(taskA * 2.8, "##")
```

```
txtEmployment(57) = Format(taskB * 1.7, "##")
```

```
txtEmployment(58) = taskD
```

```
txtEmployment(59) = Format(taskE / 5, "##")
```

```
txtEmployment(60) = Format(avgScore, "##")
```

```
Select Case CLng(avgScore)
```

```
Case 0 To 40: txtEmployment(46) = "Level 4"
```

```
Case 41 To 60: txtEmployment(46) = "Level 3"
```

```
Case 61 To 80: txtEmployment(46) = "Level 2"
```

```
Case 81 To 90: txtEmployment(46) = "Level 1"
```

```
Case 91 To 100: txtEmployment(46) = "No Service Needed"
```

```
End Select
```

```
End Sub
```

```
Private Sub lstEmployment_LostFocus(Index As Integer)
```

```
If Index = 1 Then
```

```
If CheckForTab Then
```

```
If lstEmployment(1).ListIndex <> 2 Then tabEmployment.Tab = 5
```

```
End If
```

```
ElseIf Index = 2 Then
```

```
If lstEmployment(Index).SelCount > 0 Then
```

```
If lstEmployment(Index).Selected(0) Then chkGoals(22).Value = 1 ' SSI
```

```
If lstEmployment(Index).Selected(1) Then chkGoals(23).Value = 1 ' SSDI
```

```
If lstEmployment(Index).Selected(2) Then chkGoals(24).Value = 1 ' AFDC
```

```
If lstEmployment(Index).Selected(3) Then chkGoals(25).Value = 1 ' TANF
```

```
If lstEmployment(Index).Selected(4) Then chkGoals(26).Value = 1 ' Food Stamps
```

```
If lstEmployment(Index).Selected(5) Then chkGoals(27).Value = 1 ' Section 8
```

```
If Not lstEmployment(Index).Selected(6) Then chkGoals(6).Value = 1 ' Discontinue use
```

```
entitlements - goal
```

```
End If
```

```
If CheckForTab Then
```

```
If cboEmploymentStatus(0).ListIndex <> -1 Then
```

```
If cboEmploymentStatus(0).ListIndex <> 2 Then
```

```
tabEmployment.Tab = 1
```

```
Else
```

```
tabEmployment.Tab = 4
```

```
End If
```

```
ElseIf cboEmploymentStatus(1).ListIndex <> -1 Then
```

```
tabEmployment.Tab = 4
```

```
Else
```

```
MsgBox "Please select one of the Employment statuses!", vbInformation + vbOKOnly, "Em
```

```
ployment Status"
```

```
End If
```

```
End If
```

```
End If
```

```
End Sub
```

```
Private Sub tabEmployment_Click(PreviousTab As Integer)
```

```
Dim i As Integer
```

```
If IsArrowKey > 0 Then Exit Sub
```

```
If PreviousTab = 0 Then
```

```
If cboEmploymentStatus(0).ListIndex = -1 And cboEmploymentStatus(1).ListIndex = -1 Then
```

```
MsgBox "Please select one of the Employment Statuses!", vbInformation + vbOKOnly, "Prev
```

```
Tab"
```

```

Exit Sub
ElseIf cboEmploymentStatus(0).ListIndex <> -1 Then
    If cboEmploymentStatus(0).ListIndex <> 2 Then
        'If tabEmployment.Tab > 3 And tabEmployment.Tab < 7 Then
            tabEmployment.Tab = 1
            If Not IsShftTab Then txtEmployment(9).SetFocus
            IsShftTab = False
            frmEmployment.Caption = "B. Job Retention - Employment"
        'Else
        'End If
    Else
        'If tabEmployment.Tab = 1 Or tabEmployment.Tab = 2 Or tabEmployment.Tab = 3 Then
            tabEmployment.Tab = 4
            If Not IsShftTab Then cboGeneralInfo(9).SetFocus
            IsShftTab = False
            frmEmployment.Caption = "E. Counselling, Placement && Supported - Vocational"
        'End If
    End If
Exit Sub
Else
    'If tabEmployment.Tab = 1 Or tabEmployment.Tab = 2 Or tabEmployment.Tab = 3 Then
        tabEmployment.Tab = 4
        If Not IsShftTab Then cboGeneralInfo(9).SetFocus
        IsShftTab = False
        frmEmployment.Caption = "E. Counselling, Placement && Supported - Vocational"
    'End If
End If
'ElseIf PreviousTab = 3 Then
    If cboGeneralInfo(6).ListIndex = -1 Then
        tabEmployment.Tab = 3
        cboGeneralInfo(6).SetFocus
        MsgBox "You had to select one of items from Strategy Learning!", vbInformation + vbOKO
        "Strategy Learning"
        Exit Sub
    End If

    For i = 0 To 4
        If cboFactors1(i).ListIndex = -1 Then
            tabEmployment.Tab = 3
            cboFactors1(i).SetFocus
            MsgBox "You had to select one of items from Factors!", vbInformation + vbOKOnly, "F
rs"
        Exit Sub
    End If
Next

'ElseIf PreviousTab = 8 Then
    If cboGeneralInfo(15).ListIndex = -1 Then
        tabEmployment.Tab = 8
        cboGeneralInfo(15).SetFocus
        MsgBox "You had to select one of items from Strategy Learning!", vbInformation + vbOKO
        "Strategy Learning"
        Exit Sub
    End If

    For i = 0 To cboFactors.Count - 1
        If cboFactors(i).ListIndex = -1 Then
            tabEmployment.Tab = 8
            cboFactors(i).SetFocus
            MsgBox "You had to select one of items from Factors!", vbInformation + vbOKOnly, "
rs"
        Exit Sub
    End If
Next
End If

If tabEmployment.Tab = 0 Then
    cboStaffs.SetFocus
    frmEmployment.Caption = "A. General Information"

```

```

ElseIf tabEmployment.Tab = 1 Then
    If Not IsShftTab Then txtEmployment(9).SetFocus
    IsShftTab = False
    frmEmployment.Caption = "B. Job Retention - Employment"
ElseIf tabEmployment.Tab = 2 Then
    If Not IsShftTab Then cmdInstruct.SetFocus
    IsShftTab = False
    frmEmployment.Caption = "C. Job Retention - Self Assessment"
ElseIf tabEmployment.Tab = 3 Then
    If Not IsShftTab Then cboRating(0).SetFocus
    IsShftTab = False
    frmEmployment.Caption = "D. Job Retention - Service Provider Rating, Factors And Learning
tegy"
ElseIf tabEmployment.Tab = 4 Then
    If Not IsShftTab Then cboGeneralInfo(9).SetFocus
    IsShftTab = False
    frmEmployment.Caption = "E. Counseling, Placement && Supported - Vocational"
ElseIf tabEmployment.Tab = 5 Then
    If cboEmploymentStatus(1).ListIndex = 3 Then ' If status is "not previous employed",
        tabEmployment.Tab = 6 ' skip to Consumer Self-Assessment
    frmEmployment.Caption = "G. Counseling, Placement && Supported - Consumer Self-Assessmen
cboGeneralInfo(25).SetFocus
End If
    If Not IsShftTab Then txtEmployment(25).SetFocus
    IsShftTab = False
    frmEmployment.Caption = "F. Counseling, Placement && Supported - Job History"
ElseIf tabEmployment.Tab = 6 Then
    If Not IsShftTab Then cboGeneralInfo(12).SetFocus
    IsShftTab = False
    frmEmployment.Caption = "G. Counseling, Placement && Supported - Consumer Self-Assessment"
ElseIf tabEmployment.Tab = 7 Then
    If Not IsShftTab Then cboScaleRating(9).SetFocus
    IsShftTab = False
    frmEmployment.Caption = "H. Counseling, Placement && Supported - Consumer Self-Assessment
Service Provider Rating"
ElseIf tabEmployment.Tab = 8 Then
    If Not IsShftTab Then cboGeneralInfo(15).SetFocus
    IsShftTab = False
    frmEmployment.Caption = "I. Counseling, Placement && Supported - Service Provider Rating,
ning Strategy And Factors"
ElseIf tabEmployment.Tab = 9 Then
    If Not IsGoalsClicked Then
        If Not IsShftTab Then cmdGoals.SetFocus
        IsShftTab = False
        frmEmployment.Caption = "J. Employment Services Agreement"
    End If
End If
End Sub

```

```
Private Sub txtCurrentRecord_LostFocus()
```

```

    If CheckForTab Then
        cboStaffs.SetFocus
        frmEmployment.Caption = "General Information"
    End If
End Sub

```

```
Private Sub txtEmployment_Change(Index As Integer)
```

```

    If Index = 15 Or Index = 41 Or Index = 43 Then
        If Len(txtEmployment(Index)) = 4 Then
            If Not IsNumeric(Mid(txtEmployment(Index), 2, 3)) Then
                MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, "
ne Number"
                txtEmployment(Index) = Mid(txtEmployment(Index), 1, 2)
                txtEmployment(Index).SelStart = 2
            Else
                txtEmployment(Index) = txtEmployment(Index) & "-"
                txtEmployment(Index).SelStart = 7
            End If
        End If
    End Sub

```

```

ElseIf Len(txtEmployment(Index)) = 9 Then
    If Not IsNumeric(Mid(txtEmployment(Index), 7, 3)) Then
        MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, " P
ne Number"
        txtEmployment(Index) = Mid(txtEmployment(Index), 1, 6)
        txtEmployment(Index).SelStart = 7
    Else
        txtEmployment(Index) = txtEmployment(Index) & "--"
        txtEmployment(Index).SelStart = 11
    End If
ElseIf Len(txtEmployment(Index)) = 14 Then
    If Not IsNumeric(Mid(txtEmployment(Index), 11, 3)) Then
        MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, " F
ne Number"
        txtEmployment(Index) = Mid(txtEmployment(Index), 1, 10)
        txtEmployment(Index).SetFocus
    End If
End If
End If
End Sub

```

```

Private Sub txtEmployment_GotFocus(Index As Integer)

```

```

    If Index = 25 Then ' Most recent job
        If cboEmploymentStatus(1).ListIndex = 3 Then ' "Not previously employed" for employment s
us 2
            cboGeneralInfo(12).SetFocus ' Skip to Consumer Self-Assessment Questions
            chkGoals(3).Value = 1 ' complete an internship checkbox
        End If
    End If
End Sub

```

```

If Index = 15 Or Index = 41 Or Index = 43 Then
    txtEmployment(Index) = "("
    txtEmployment(Index).SelStart = 2
End If
End Sub

```

```

Private Sub txtEmployment_LostFocus(Index As Integer)
    Dim tmpStr As String
    Dim tmpMonth As Long
    Dim tmpYear As Long

```

```

    If Index = 10 Then
        If CheckForShiftTab Then
            IsShiftTab = True
            tabEmployment.Tab = 0
        End If
    ElseIf Index = 15 Or Index = 41 Or Index = 43 Then
        If txtEmployment(Index) = "(" Then txtEmployment(Index) = ""
    ElseIf Index = 16 Then
        If CheckForTab Then
            tabEmployment.Tab = 2
        End If
    ElseIf Index = 24 Then
        If CheckForTab Then
            tabEmployment.Tab = 5
        End If
    ElseIf Index = 26 Then
        If CheckForShiftTab Then
            IsShiftTab = True
            tabEmployment.Tab = 4
            If lstEmployment(1).ListIndex = 2 Then
                lstEmployment(1).SetFocus
            Else
                txtEmployment(25).SetFocus
            End If
        End If
    ElseIf Index = 29 Then
        If Trim(txtEmployment(Index)) <> "" Then
            tmpMonth = DateDiff("m", Trim(txtEmployment(Index)), Now)

```

```

    tmpYear = tmpMonth / 12
    tmpMonth = tmpMonth Mod 12
    If tmpYear > 1 Then
        tmpStr = tmpYear & " years"
        chkGoals(3).Value = 1
    ElseIf tmpYear = 1 Then
        tmpStr = tmpYear & " year"
    End If

    If tmpYear > 0 Then
        If tmpMonth > 1 Then
            tmpStr = tmpStr & " and " & tmpMonth & " monthes"
        ElseIf tmpMonth = 1 Then
            tmpStr = tmpStr & " and " & tmpMonth & " month"
        End If
    Else
        If tmpMonth > 1 Then
            tmpStr = tmpMonth & " monthes"
        ElseIf tmpMonth = 1 Then
            tmpStr = tmpMonth & " month"
        End If
    End If

    txtEmployment(30) = tmpStr
End If
ElseIf Index = 38 Then
    If CheckForTab Then
        tabEmployment.Tab = 6
    End If
End If
End Sub

Private Sub txtEmployment_Validate(Index As Integer, Cancel As Boolean)

    If Index = 1 Or Index = 7 Or Index = 8 Or Index = 10 Or Index = 21 Or Index = 22 Or _
        Index = 27 Or Index = 33 Or Index = 37 Then
        If Trim(txtEmployment(Index)) <> "" Then
            If Not IsNumeric(txtEmployment(Index)) Then
                MsgBox "Please enter numeric number in this fields", vbInformation + vbOKOnly, "Nume
Number"
                txtEmployment(Index) = ""
                Cancel = True
            End If
        End If
    ElseIf Index = 0 Or Index = 2 Or Index = 29 Or Index = 50 Then
        If txtEmployment(Index) <> "" Then
            If Not IsDate(Trim(txtEmployment(Index))) Then
                MsgBox "Please enter the date into this field!", vbInformation + vbOKOnly, "Date"
                txtEmployment(Index) = ""
                Cancel = True
            ElseIf DateValue(Trim(txtEmployment(Index))) - DateValue(Date) > 0 Then
                MsgBox "Please enter the date into this field not later than Today!", vbInformation
OKOnly, "Date"
                txtEmployment(Index) = ""
                Cancel = True
            End If
        End If
    End If
End Sub

Private Sub CountGoals()
Dim i As Integer

For i = 0 To cboAgree.Count - 1
    Select Case i
        Case 0 To 1
            If cboAgree(i).Text = "Yes" Then chkGoals(0) = 1
        Case 2 To 4
            If cboAgree(i).Text = "Yes" Then chkGoals(1) = 1
        Case 5 To 7
    
```

```
        If cboAgree(i).Text = "Yes" Then chkGoals(2) = 1
    Case 8
        If cboAgree(i).Text = "Yes" Then chkGoals(3) = 1
    Case 9 To 11
        If cboAgree(i).Text = "Yes" Then chkGoals(4) = 1
    Case 12 To 13
        If cboAgree(i).Text = "Yes" Then chkGoals(5) = 1
    Case 14 To 19
        If cboAgree(i).Text = "Yes" Then chkGoals(6) = 1
    End Select
Next
End Sub
```

```
Option Explicit
Public IsNewRecord As Boolean
Private AgencyName As String
Private rstQA As ADODB.Recordset
Private rsAgency As ADODB.Recordset
```

```
Private Sub cmdClose_Click()
Unload Me
End Sub
```

```
Private Sub cmdOverallScore_Click()
frmOverallScore.Show vbModal
End Sub
```

```
Private Sub cmdSave_Click()
```

```
SaveRecords
```

```
End Sub
```

```
Private Sub SaveRecords()
Dim tmpParticipantNumber As String
Dim tmpInterviewDate As Date
Dim tmpStaff As String
Dim tmpFrequency1 As String
Dim tmpFrequency2 As String
Dim tmpFrequency3 As String
Dim tmpFrequency4 As String
Dim tmpFrequency5 As String
Dim tmpSatisfaction As String
Dim tmpActivity As String
Dim tmpFrequencyTwo As String
Dim tmpEffect As String
Dim EndDate1 As Date
Dim tmpStr As String
```

```
' Mandatory fields
```

```
If Trim(txtParticipantNumber) = "" Then
MsgBox "You must enter a Participant Number!", vbOKOnly, "Save"
txtParticipantNumber.SetFocus
Exit Sub
End If
```

```
If Trim(txtInterviewDate) = "" Then
MsgBox "You must enter an Interview Date!", vbOKOnly, "Save"
txtInterviewDate.SetFocus
Exit Sub
End If
```

```
If cboStaffs.ListIndex = -1 Then
MsgBox "You must enter a Staff Name!", vbOKOnly, "Save"
cboStaffs.SetFocus
Exit Sub
Else
tmpStaff = cboStaffs.ItemData(cboStaffs.ListIndex)
End If
```

```
If cboActivity.ListIndex = -1 Then
MsgBox "You must answer the activity level question!", vbOKOnly, "Save"
cboActivity.SetFocus
Exit Sub
Else
tmpActivity = cboActivity.ItemData(cboActivity.ListIndex)
End If
```

```
If cboFrequency2.ListIndex = -1 Then
MsgBox "You must answer the social integration question!", vbOKOnly, "Save"
cboFrequency2.SetFocus
Exit Sub
```

```

Else
    tmpFrequencyTwo = cboFrequency2.ItemData(cboFrequency2.ListIndex)
End If

If cboEffect.ListIndex = -1 Then
    MsgBox "You must answer the function independently question!", vbOKOnly, "Save"
    cboEffect.SetFocus
    Exit Sub
Else
    tmpEffect = cboEffect.ItemData(cboEffect.ListIndex)
End If

If cboFrequency1(0).ListIndex = -1 Then
    tmpFrequency1 = ""
Else
    tmpFrequency1 = cboFrequency1(0).ItemData(cboFrequency1(0).ListIndex)
End If

If cboFrequency1(1).ListIndex = -1 Then
    tmpFrequency2 = ""
Else
    tmpFrequency2 = cboFrequency1(1).ItemData(cboFrequency1(1).ListIndex)
End If

If cboFrequency1(2).ListIndex = -1 Then
    tmpFrequency3 = ""
Else
    tmpFrequency3 = cboFrequency1(2).ItemData(cboFrequency1(2).ListIndex)
End If

If cboFrequency1(3).ListIndex = -1 Then
    tmpFrequency4 = ""
Else
    tmpFrequency4 = cboFrequency1(3).ItemData(cboFrequency1(3).ListIndex)
End If

If cboFrequency1(4).ListIndex = -1 Then
    tmpFrequency5 = ""
Else
    tmpFrequency5 = cboFrequency1(4).ItemData(cboFrequency1(4).ListIndex)
End If

If cboSatisfaction.ListIndex = -1 Then
    tmpSatisfaction = ""
Else
    tmpSatisfaction = cboSatisfaction.ItemData(cboSatisfaction.ListIndex)
End If

If Trim(txtEndDate1) = "" Then
    EndDate1 = "12:00:00 PM"
Else
    EndDate1 = txtEndDate1
End If

If (rstQA.BOF = True And rstQA.EOF = True) Then
    cnn.Execute "INSERT INTO QA (ConsumerID, Agency, Episode, CaseNumber, ParticipantNumber, " & _
        "InterviewDate, Interviewer, Appointments, Needs, Helpful, Respect, Decisions, " & _
        "Satisfaction, Activity, Social, Functioning, " & _
        "EndDate1) VALUES (" & CustomerID & ", " & AgencyCode & ", " & Episode & ", " & _
        CaseNumber & ", " & _
        txtParticipantNumber & ", " & txtInterviewDate & ", " & tmpStaff & ", " & _
        Frequency1 & ", " & _
        tmpFrequency2 & ", " & tmpFrequency3 & ", " & tmpFrequency4 & ", " & tmp
        ency5 & ", " & _
        tmpSatisfaction & ", " & tmpActivity & ", " & tmpFrequencyTwo & ", " & _
        tmpEffect & ", " & EndDate1 & ")"
Else

```



```

cnn.Execute "UPDATE QA Set ParticipantNumber = '" & txtParticipantNumber & "', InterviewDate = '" & txtInterviewDate & _
            "'", Interviewer = '" & tmpStaff & "', Appointments = '" & tmpFrequency1 & "', Ne
            = '" & tmpFrequency2 & _
            "'", Helpful = '" & tmpFrequency3 & "', Respect = '" & tmpFrequency4 & "', Decis:
            = '" & tmpFrequency5 & _
            "'", Satisfaction = '" & tmpSatisfaction & "', Activity = '" & tmpActivity & _
            "'", Social = '" & tmpFrequencyTwo & "', Functioning = '" & tmpEffect & _
            "'", EndDate1 = '" & EndDate1 & "' WHERE ConsumerID = '" & CustomerID & "' AND Age:
= '" & AgencyCode & _
            "' AND Episode = '" & Episode & "' AND CaseNumber = '" & CaseNumber & "'
End If

```

```
cmdSave.Enabled = False
```

```
MsgBox "End of Episode QA successfully saved!", vbOKOnly, "Save"
```

```
Exit Sub
```

```
errSave:
```

```
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
```

```
End Sub
```

```
Private Sub Form_Load()
```

```

txtCaseNumber.Text = CaseNumber
txtEpisode.Text = Episode
txtConsumerName.Text = FirstName & " " & LastName
txtPhoneNumber = PhoneNo

```

```
AgencyName = frmMainMenu.txtAgencyName
```

```

lblAppointments.Caption = "The following questions are regarding " & AgencyName & " services
eneral. Were your appointments made quickly?"
lblNeeds.Caption = "Did the " & AgencyName & " staff understand your needs?"
lblDecisions.Caption = "How often were you involved in decisions about the services you recei
at " & AgencyName & "? "
lblSatisfied.Caption = "Overall, how satisfied were you with " & AgencyName & " services? Wo
you say you were? "
lblFunctioning.Caption = "How much have " & AgencyName & " services affected your ability to
tion independently?"

```

```

LoadControlsItems cboStaffs, False
LoadControlsItems cboFrequency1, True
LoadControlsItems cboSatisfaction, False
LoadControlsItems cboActivity, False
LoadControlsItems cboFrequency2, False
LoadControlsItems cboEffect, False

```

```

Set rstQA = New Recordset
rstQA.CursorLocation = adUseClient

```

```

rstQA.Open "SELECT * FROM QA WHERE ConsumerId = '" & CustomerID & "' AND Episode = '" & Episode
" AND Agency = '" & AgencyCode & "' AND CaseNumber = '" & CaseNumber & "'", cnn,
penDynamic, adLockOptimistic

```

```
With rstQA
```

```

    If Not (.EOF And .BOF) Then
        .MoveFirst
    
```

```
LoadDataToControls rstQA, txtParticipantNumber, False
```

```

    If ([InterviewDate] = "12:00:00 PM" Or Trim([InterviewDate] & vbNullString) = "") Then
        txtInterviewDate = ""
    Else
        txtInterviewDate = [InterviewDate]
    End If

```

```

LoadDataToControls rstQA, cboStaffs, False
LoadDataToControls rstQA, cboFrequency1, True

```

```
LoadDataToControls rstQA, cboSatisfaction, False
LoadDataToControls rstQA, cboActivity, False
LoadDataToControls rstQA, cboFrequency2, False
LoadDataToControls rstQA, cboEffect, False
```

```
If ![EndDate1] = "12:00:00 PM" Or Not IsDate(![EndDate1]) Then
    txtEndDate1 = ""
    cmdOverallScore.Enabled = False
Else
    txtEndDate1 = ![EndDate1]
End If
```

```
If ![EndDate1] <> "12:00:00 PM" Or IsDate(![EndDate1]) Then
    cmdSave.Enabled = False
    cmdOverallScore.Enabled = True
End If
```

```
End If
End With
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    rstQA.Close
```

```
End Sub
```

```

Option Explicit
Public IsNewRecord As Boolean
Private rstIntervention As ADODB.Recordset
Private InterventionSeq As Long

Private Sub cboExtent_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
If KeyCode = vbKeyBack Then
    cboExtent(Index).ListIndex = -1
End If
End Sub

Private Sub cmdClose_Click()
Unload Me
Set rstIntervention = Nothing
End Sub

Private Sub cmdSave_Click()
Dim x As Integer
Dim Learning As String
Dim Techniques As String
Dim Responsibility As String
Dim Realistic As String

On Error GoTo errSave

For x = 0 To cboExtent.Count - 1
    If cboExtent(x).ListIndex = -1 Then
        MsgBox "Please answer all questions to save the report", vbOKOnly, "Save"
        cboExtent(x).SetFocus
        Exit Sub
    End If
Next

Learning = cboExtent(0).ItemData(cboExtent(0).ListIndex)
Techniques = cboExtent(1).ItemData(cboExtent(1).ListIndex)
Responsibility = cboExtent(2).ItemData(cboExtent(2).ListIndex)
Realistic = cboExtent(3).ItemData(cboExtent(3).ListIndex)

cnn.Execute "UPDATE Interventions SET LearningProcess = '" & Learning & "'," & _
    " Techniques = '" & Techniques & "'," & _
    " Responsibility = '" & Responsibility & "'," & _
    " Realistic = '" & Realistic & "' & _
    " WHERE ConsumerId = '" & CustomerID & _
    " AND Agency = '" & AgencyCode & "' & _
    " AND CaseNumber = '" & CaseNumber & "' & _
    " AND InterventionID = '" & frmIntervention1.InterventionCode & "' & _
    " AND InterventionSequence = '" & InterventionSeq

MsgBox "Successfully Saved!", vbOKOnly, "EndOfService"
Exit Sub

errSave:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub

Private Sub Form_Load()
txtConsumerName.Text = frmIntervention1.txtFirstName & " " & frmIntervention1.txtLastName
txtCaseNumber.Text = frmIntervention1.txtCaseNumber
txtEpisode.Text = frmIntervention1.txtEpisode
txtIntervention = frmIntervention1.txtIntervention
InterventionSeq = frmIntervention1.SequenceNo

LoadControlsItems cboExtent, True

Set rstIntervention = New ADODB.Recordset

rstIntervention.Open "SELECT LearningProcess, Techniques, Responsibility, Realistic" & _
    " FROM Interventions WHERE ConsumerId = '" & CustomerID & _
    " AND Agency = '" & AgencyCode & "' AND CaseNumber = '" & CaseNumber &

```

```
    " AND InterventionID = '" & frmIntervention1.InterventionCode & "' & _  
    " AND InterventionSequence = '" & interventionSeq, cnn, adOpenDynamic, ad
```

Optimistic

LoadDataToControls rstIntervention, cboExtent, True

rstIntervention.Close  
End Sub

```
Option Explicit
Public IsNewRecord As Boolean
Public IsFileDateChecked As Boolean
Private rstEyeData As ADODB.Recordset
```

```
Private Sub cboAcuity_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboAcuity(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboAcuitySource_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboAcuitySource(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboDiagnosis_Change(Index As Integer)
Dim j As Integer
```

```
If IsNumeric(Trim(cboDiagnosis(Index).Text)) And Len(Trim(cboDiagnosis(Index).Text)) = 3 Then
    For j = 27 To UBound(DiagnosisArray)
        If Trim(cboDiagnosis(Index).Text) <= Mid(DiagnosisArray(j), 1, 3) Then
            cboDiagnosis(Index).ListIndex = j
            Exit Sub
        End If
    Next
ElseIf Trim(cboDiagnosis(Index).Text) <> "" And Len(Trim(cboDiagnosis(Index).Text)) = 1 Then
    If (Asc(Trim(cboDiagnosis(Index).Text)) > 64 And Asc(Trim(cboDiagnosis(Index).Text)) < 91)
        Then
            (Asc(Trim(cboDiagnosis(Index).Text)) > 96 And Asc(Trim(cboDiagnosis(Index).Text)) < 123)
            For j = 0 To 24
                If Asc(UCase(Trim(cboDiagnosis(Index).Text))) <= Asc(Mid(DiagnosisString(j), 1, 1))
                    Then
                        cboDiagnosis(Index).ListIndex = j
                        Exit Sub
                    End If
            Next
        End If
    End If
End Sub
```

```
Private Sub cboDiagnosis_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboDiagnosis(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboLengthAddress_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboLengthAddress.ListIndex = -1
End If
End Sub
```

```
Private Sub cboStaffs_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboStaffs.ListIndex = -1
End If
End Sub
```

```
Private Sub cboVisualSource_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboVisualSource.ListIndex = -1
End If
```

```
End If
End Sub
```

```
Private Sub cmdClear_Click()
```

```
IsNewRecord = True
clearData cboStaffs, False
clearData txtEyeData, True
clearData cboDiagnosis, True
clearData lstHealthCondition, False
clearData cboDiagnosis, True
clearData cboAcuity, True
clearData cboVisualSource, False
clearData cboAcuitySource, True
clearData cboLengthAddress, False
clearData chkBox, True
clearData txtGlass, True
clearData cboVisualField, True
```

```
cmdSave.Enabled = True
End Sub
```

```
Private Sub cmdClose_Click()
Unload Me
End Sub
```

```
Private Sub cmdNextRecord_Click()
```

```
With rstEyeData
.MoveNext
If Not .EOF Then
txtCurrentRecord = .AbsolutePosition
If .RecordCount = .AbsolutePosition Then cmdNextRecord.Enabled = False
cmdPreviousRecord.Enabled = True
LoadDataToControls rstEyeData, cboStaffs, False
LoadDataToControls rstEyeData, txtEyeData, True
LoadDataToControls rstEyeData, cboDiagnosis, True
LoadDataToControls rstEyeData, cboVisualSource, False
LoadDataToControls rstEyeData, cboAcuitySource, True
LoadDataToControls rstEyeData, cboAcuity, True
LoadDataToControls rstEyeData, lstHealthCondition, False
LoadDataToControls rstEyeData, cboLengthAddress, False
LoadDataToControls rstEyeData, chkBox, True
LoadDataToControls rstEyeData, txtGlass, True
If IsDate(txtEyeData(0)) Then
cmdSave.Enabled = False
Else
cmdSave.Enabled = True
End If
End If
End With
End Sub
```

```
Private Sub cmdPreviousRecord_Click()
```

```
With rstEyeData
.MovePrevious
If Not .BOF Then
txtCurrentRecord = .AbsolutePosition
If .AbsolutePosition = 1 Then cmdPreviousRecord.Enabled = False
cmdNextRecord.Enabled = True
LoadDataToControls rstEyeData, cboStaffs, False
LoadDataToControls rstEyeData, txtEyeData, True
LoadDataToControls rstEyeData, cboDiagnosis, True
LoadDataToControls rstEyeData, cboVisualSource, False
LoadDataToControls rstEyeData, cboAcuitySource, True
LoadDataToControls rstEyeData, cboAcuity, True
LoadDataToControls rstEyeData, lstHealthCondition, False
LoadDataToControls rstEyeData, cboLengthAddress, False
LoadDataToControls rstEyeData, chkBox, True

```

```
LoadDataToControls rstEyeData, txtGlass, True
If IsDate(txtEyeData(0)) Then
    cmdSave.Enabled = False
Else
    cmdSave.Enabled = True
End If
End If
End With
End Sub
```

```
Private Sub cmdSave_Click()
Dim colEyeData As New Collection
```

```
On Error GoTo errSave
```

```
colEyeData.Add cboStaffs
colEyeData.Add cboDiagnosis
colEyeData.Add cboVisualSource
colEyeData.Add cboAcuitySource
colEyeData.Add cboAcuity
colEyeData.Add txtEyeData
colEyeData.Add lstHealthCondition
colEyeData.Add cboLengthAddress
colEyeData.Add chkBox
colEyeData.Add txtGlass
```

```
SaveDataToDB frmEyeData, rstEyeData, colEyeData, "0", False 'Call public function to
e data to DB
MsgBox "Successfully saved the Assessment information!", vbOKOnly, "Save"
```

```
IsNewRecord = False
Set colEyeData = Nothing
Exit Sub
```

```
errSave:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub
```

```
Private Sub Form_Load()
Dim IndexNo As Integer
Dim i As Integer, j As Integer, k As Integer
```

```
On Err GoTo Errhandler
```

```
txtFirstName = FirstName & " " & LastName
txtAddress = Address
txtCaseNumber = CaseNumber
```

```
LoadControlsItems cboStaffs, False
```

```
Set rstEyeData = New ADODB.Recordset
rstEyeData.CursorLocation = adUseClient
rstEyeData.Open "Select Code, Description From Diagnosis", cnn, adOpenForwardOnly, adLockRead
With rstEyeData
    While Not .EOF
        cboDiagnosis(0).AddItem ![Description]
        cboDiagnosis(1).AddItem ![Description]
        cboDiagnosis(0).ItemData(cboDiagnosis(0).NewIndex) = IndexNo
        cboDiagnosis(1).ItemData(cboDiagnosis(1).NewIndex) = IndexNo
        DiagnosisArray(IndexNo) = ![Code]
        If IndexNo < 25 Then DiagnosisString(IndexNo) = ![Description]
        IndexNo = IndexNo + 1
        .MoveNext
    Wend
    cboDiagnosis(0).ListIndex = -1
    cboDiagnosis(1).ListIndex = -1
End With
rstEyeData.Close
```

```

IndexNo = 0
rstEyeData.Open "Select Code, Description From VisualField", cnn, adOpenForwardOnly, adLockRead
ly
With rstEyeData
    While Not .EOF
        cboVisualField(0).AddItem ![Description]
        cboVisualField(1).AddItem ![Description]
        cboVisualField(0).ItemData(cboVisualField(0).NewIndex) = IndexNo
        cboVisualField(1).ItemData(cboVisualField(1).NewIndex) = IndexNo
        VisualFieldArray(IndexNo) = ![Code]
        IndexNo = IndexNo + 1
        .MoveNext
    Wend
End With
rstEyeData.Close

rstEyeData.Open "Select * From Eyedata Where ConsumerID = " & CustomerID & " And Agency = '" &
    AgencyCode & "' And CaseNumber = '" & CaseNumber & "' Order By SequenceNo DESC", cnn, a
penDynamic, adLockOptimistic
With rstEyeData
    If Not (.BOF And .EOF) Then
        txtRecordNo = .RecordCount
        txtCurrentRecord = 1
        If .RecordCount > 1 Then cmdNextRecord.Enabled = True
        LoadDataToControls rstEyeData, cboStaffs, False
        LoadDataToControls rstEyeData, txtEyeData, True
        LoadDataToControls rstEyeData, cboDiagnosis, True
        LoadDataToControls rstEyeData, cboVisualSource, False
        LoadDataToControls rstEyeData, cboAcuitySource, True
        LoadDataToControls rstEyeData, cboAcuity, True
        LoadDataToControls rstEyeData, lstHealthCondition, False
        LoadDataToControls rstEyeData, cboLengthAddress, False
        LoadDataToControls rstEyeData, chkBox, True
        LoadDataToControls rstEyeData, txtGlass, True
        If IsDate(txtEyeData(0)) Then cmdSave.Enabled = False
    End If
End With
Exit Sub

Errhandler:
MsgBox.Err.Description, vbCritical + vbOKOnly, "Form Load"
End Sub

Private Sub Form_Unload(Cancel As Integer)

If rstEyeData.State = adStateOpen Then rstEyeData.Close
Set rstEyeData = Nothing
Set frmEyeData = Nothing
End Sub

Private Sub lstHealthCondition_KeyUp(KeyCode As Integer, Shift As Integer)
Dim i As Integer

If KeyCode = vbKeyBack Then
    If lstHealthCondition.SelCount = 1 Then
        lstHealthCondition.Selected(lstHealthCondition.ListIndex) = False
    ElseIf lstHealthCondition.ListCount > 1 Then
        For i = lstHealthCondition.ListCount - 1 To 0 Step -1
            If lstHealthCondition.Selected(i) Then
                lstHealthCondition.Selected(i) = False
            End If
        Next
    End If
End If
End Sub

Private Sub txtEyeData_Validate(Index As Integer, Cancel As Boolean)

If Index = 0 Then
    If txtEyeData(Index) <> "" Then

```



mEyeData - 5

60439706.0113C

```
If Not IsDate(Trim(txtEyeData(Index))) Then
    MsgBox "Please enter date in this field", vbInformation + vbOKOnly, "Assessment Date"
    txtEyeData(Index) = ""
    ~Cancel = True
End If
End If
End If
End Sub
```

```
Option Explicit
Private rst As ADODB.Recordset      'resordset object variable
Private rstILAssessment As ADODB.Recordset
Public IsNewRecord As Boolean
Private IsArrowKey As Integer
Private IsGoalsClicked As Boolean
Public IsFileDateChecked As Boolean
Private IsVisualInfoClicked As Boolean

Private Sub cboAgencyServices_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboAgencyServices(Index).ListIndex = -1
    End If
End Sub

Private Sub cboAgree_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboAgree(Index).ListIndex = -1
    End If
End Sub

Private Sub cboAgree_LostFocus(Index As Integer)
    If CheckForTab Then
        If Index = 1 Then
            tabILAssessment.Tab = 1
        ElseIf Index = 19 Then
            tabILAssessment.Tab = 2
        ElseIf Index = 31 Then
            tabILAssessment.Tab = 3
        ElseIf Index = 50 Then
            tabILAssessment.Tab = 5
        ElseIf Index = 73 Then
            tabILAssessment.Tab = 7
        End If
    End If
End Sub

Private Sub cboBrailleLevel_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboBrailleLevel(Index).ListIndex = -1
    End If
End Sub

Private Sub cboDifficulty_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboDifficulty(Index).ListIndex = -1
    End If
End Sub

Private Sub cboEnlargerName_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboEnlargerName.ListIndex = -1
    End If
End Sub

Private Sub cboEnlargerName_LostFocus()
    If CheckForTab Then
        If cboEnlargerName.ListIndex <> 2 Then cboAgree(13).SetFocus
    End If
End Sub

Private Sub cboEquipment_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboEquipment(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboExternalServices_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboExternalServices(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboFactors_Click(Index As Integer)
```

```
If IsArrowKey > 0 Then Exit Sub
If Index = 5 Then
    CountScore
    CountGoals
End If
End Sub
```

```
Private Sub cboFactors_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub
```

```
Private Sub cboFactors_LostFocus(Index As Integer)
```

```
If CheckForTab Then
    If Index = 5 Then
        CountScore
        CountGoals
    End If
End If
```

```
If IsArrowKey > 0 Then IsArrowKey = 0
End Sub
```

```
Private Sub cboLargerLVDeviceSize_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboLargerLVDeviceSize(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboObserveResponse_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboObserveResponse(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboObserveResponse_LostFocus(Index As Integer)
```

```
If CheckForTab Then
    If Index = 7 Then
        tabILAssessment.Tab = 8
    End If
End If
End Sub
```

```
Private Sub cboRating_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboRating(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboRating_LostFocus(Index As Integer)
```

```
If CheckForTab Then
    If Index = 35 Then
        tabILAssessment.Tab = 4
    ElseIf Index = 14 Then
        tabILAssessment.Tab = 6
    End If
End If
End Sub
```

```
Private Sub cboReaderName_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboReaderName.ListIndex = -1
End If
End Sub
```

```
Private Sub cboReaderName_LostFocus()
```

```
If CheckForTab Then
    If cboReaderName.ListIndex <> 2 Then cboAgree(14).SetFocus
End If
End Sub
```

```
Private Sub cboServiceLevel_Click(Index As Integer)
```

```
If IsArrowKey > 0 Then Exit Sub
```

```
If Index = 0 Then
    If cboServiceLevel(Index).ListIndex = -1 Then
        txtILAssessment(15) = ""
    Else
        cboTimeFrame.ListIndex = cboServiceLevel(Index).ListIndex - 1
        txtILAssessment(15) = cboTimeFrame.Text
    End If

```

```
    tabILAssessment.Tab = 10
    Call CountGoals
End If
```

```
End Sub
```

```
Private Sub cboServiceLevel_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub
```

```
Private Sub cboServiceLevel_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
    cboServiceLevel(Index).ListIndex = -1
End If
End Sub
```

```
Private Sub cboServiceLevel_LostFocus(Index As Integer)
```

```
If Index = 0 Then
  If CheckForTab Then
    If cboServiceLevel(Index).ListIndex = 0 Then
      txtILAssessment(15) = ""
    Else
      If cboServiceLevel(Index).ListIndex <> -1 Then
        cboTimeFrame.ListIndex = cboServiceLevel(Index).ListIndex - 1
        txtILAssessment(15) = cboTimeFrame.Text
      End If
    End If

    tabILAssessment.Tab = 10
    Call CountGoals
    cmdInstruction(4).SetFocus
  End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboStaffs_KeyUp(KeyCode As Integer, Shift As Integer)

  If KeyCode = vbKeyBack Then
    cboStaffs.ListIndex = -1
  End If
End Sub

Private Sub cboYesNo_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

  If KeyCode = vbKeyBack Then
    cboYesNo(Index).ListIndex = -1
  End If
End Sub

Private Sub cboYesNo_LostFocus(Index As Integer)

  If Index = 8 Then
    If CheckForTab Then
      tabILAssessment.Tab = 9
      cboStrategiesLearning.SetFocus
      frmILAssessment.Caption = "Strategies Learning And Factors"
    End If
  ElseIf Index = 10 Then
    If CheckForTab Then
      If cboYesNo(Index).ListIndex = 1 Then cboYesNo(8).SetFocus
    End If
  ElseIf Index >= 11 And Index <= 29 Then
    If CheckForTab Then
      If cboYesNo(Index).ListIndex = 0 Then
        cboObserverResponse(Index - 11).SetFocus
      Else
        If Index = 11 Then
          txtILAssessment(4).SetFocus
        Else
          cboYesNo(Index - 1).SetFocus
        End If
      End If
    End If
  ElseIf Index = 31 Then
    If CheckForTab Then
      If cboYesNo(31).ListIndex <> 0 Then cboYesNo(33).SetFocus
    End If
  End If
End Sub

Private Sub cmdClear_Click()
  IsNewRecord = True
  clearData txtILAssessment, True
  clearData chkGoals, True
End Sub
```

```

clearData cboYesNo, True
clearData cboAgree, True
clearData cboExternalServices, True
clearData cboAgencyServices, True
clearData cboEquipment, True
clearData cboServiceLevel, True
clearData cboFactors, True
clearData cboStrategiesLearning, False
clearData cboObserveResponse, True
clearData cboRating, True
clearData cboDifficulty, True
clearData cboStaffs, False
clearData cboLargerLVDeviceSize, True
clearData cboBrailleLevel, True
clearData cboEnlargerName, False
clearData cboReaderName, False

```

```

If rstILAssessment.State = adStateOpen Then rstILAssessment.Close
rstILAssessment.Open "Select * From ILAssessment Where ConsumerID = " & CustomerID & _
    " And Agency = " & AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & _
    CaseNumber & " And InterventionID = '013' Order By EntryDate, FinishDate ", cnn, _
    adOpenKeyset, adLockOptimistic
If Not rstILAssessment.EOF Then rstILAssessment.MoveFirst

```

```

cmdPrint.Enabled = False
cmdSave.Enabled = True
tabILAssessment.Tab = 0
End Sub

```

```

Private Sub cmdClose_Click()
Unload Me
End Sub

```

```

Private Sub cmdHistorical_Click()
IsVisualInfoClicked = True
frmEyeData.Show vbModal
End Sub

```

```

Private Sub cmdInstruction_Click(Index As Integer)

```

```

If Index = 0 Then
    MsgBox "If the answer to any consumer self-rating question is a 1 or 2:" & Chr(13) & _
        Chr(10) & "    recommend training." & Chr(13) & Chr(10) & "If 3 or 4:" & Chr(13) & Chr(10) & _
        "    request a demonstration." & Chr(13) & Chr(10) & "If consumer doesn't not answer _
        indicates they are uninterested in performing a task:" & Chr(13) & Chr(10) & "    mark as R refused", vbOKOnly, "Consumer Self-Rating."
ElseIf Index = 1 Then
    MsgBox "Rating Scale - Observe and Train:" & Chr(13) & Chr(10) & _
        "    1 = major problem, 10 = no problem, N/A, 1 = Refuses", vbOKOnly, "Rating Scale"
ElseIf Index = 2 Then
    MsgBox "If problem affects safety, indicates problem, inform consumer and record their re _
        e", vbOKOnly
ElseIf Index = 3 Then

```

```

End If
End Sub

```

```

Private Sub cmdNext_Click()

```

```

If tabILAssessment.Tab = 10 Then
    MsgBox "It is last page for Services Agreement already!", vbInformation + vbOKOnly, "Next"
Else
    tabILAssessment.Tab = tabILAssessment.Tab + 1
    frmILAssessment.Caption = getTabCaption(tabILAssessment.Tab)
End If
End Sub

```

```
Private Sub cmdNextRecord_Click()  
Dim tmpLong As Long
```

```
With rstILAssessment
```

```
    .MoveNext
```

```
    If .AbsolutePosition = 1 Then
```

```
        cmdPreviousRecord.Enabled = False
```

```
        cmdNextRecord.Enabled = True
```

```
    ElseIf .AbsolutePosition < .RecordCount Then
```

```
        cmdPreviousRecord.Enabled = True
```

```
        cmdNextRecord.Enabled = True
```

```
    ElseIf .RecordCount = .AbsolutePosition Then
```

```
        cmdPreviousRecord.Enabled = True
```

```
        cmdNextRecord.Enabled = False
```

```
    End If
```

```
    txtCurrentRecord = .AbsolutePosition
```

```
    txtSequencyNo = ![AssessmentSequence]
```

```
    LoadDataToControls rstILAssessment, cboStrategiesLearning, False
```

```
    LoadDataToControls rstILAssessment, cboStaffs, False
```

```
    LoadDataToControls rstILAssessment, txtILAssessment, True
```

```
    LoadDataToControls rstILAssessment, chkGoals, True
```

```
    LoadDataToControls rstILAssessment, cboExternalServices, True
```

```
    LoadDataToControls rstILAssessment, cboAgencyServices, True
```

```
    LoadDataToControls rstILAssessment, cboEquipment, True
```

```
    LoadDataToControls rstILAssessment, cboServiceLevel, True
```

```
    LoadDataToControls rstILAssessment, cboObserveResponse, True
```

```
    LoadDataToControls rstILAssessment, cboRating, True
```

```
    LoadDataToControls rstILAssessment, cboDifficulty, True
```

```
    LoadDataToControls rstILAssessment, cboTimeFrame, False
```

```
    LoadDataToControls rstILAssessment, cboFactors, True
```

```
    If rst.State = adStateOpen Then rst.Close
```

```
    rst.Open "Select * From ILAssessment1 Where ConsumerID = " & CustomerID & " And Agency = "
```

```
        AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & "
```

```
        " And InterventionID = '013' And AssessmentSequence = " & ![AssessmentSequence] & "
```

```
        " Order By EntryDate", cnn, adOpenKeyset, adLockOptimistic
```

```
    LoadDataToControls rst, cboYesNo, True
```

```
    LoadDataToControls rst, cboAgree, True
```

```
    rst.Close
```

```
    If ![FinishDate] = "12:00:00 PM" Then
```

```
        cmdSave.Enabled = True
```

```
        cmdPrint.Enabled = False
```

```
    Else
```

```
        cmdSave.Enabled = False
```

```
        cmdPrint.Enabled = True
```

```
    End If
```

```
End With
```

```
tabILAssessment.Tab = 0
```

```
txtCurrentRecord.SetFocus
```

```
End Sub
```

```
Private Sub cmdPrevious_Click()
```

```
    If tabILAssessment.Tab = 0 Then
```

```
        MsgBox "It is first page for Self Assessment already!", vbInformation + vbOKOnly, "Previous  
page"
```

```
    Else
```

```
        tabILAssessment.Tab = tabILAssessment.Tab - 1
```

```
        frmILAssessment.Caption = getTabCaption(tabILAssessment.Tab)
```

```
    End If
```

```
End Sub
```

```
Private Sub cmdPreviousRecord_Click()
```

```
With rstILAssessment
```

```

.MovePrevious
If .AbsolutePosition = 1 Then
    cmdPreviousRecord.Enabled = False
    cmdNextRecord.Enabled = True
ElseIf .AbsolutePosition < .RecordCount Then
    cmdPreviousRecord.Enabled = True
    cmdNextRecord.Enabled = True
ElseIf .RecordCount = .AbsolutePosition Then
    cmdPreviousRecord.Enabled = True
    cmdNextRecord.Enabled = False
End If

If ![FinishDate] = "12:00:00 PM" Then
    cmdSave.Enabled = True
    cmdPrint.Enabled = False
Else
    cmdSave.Enabled = False
    cmdPrint.Enabled = True
End If

txtCurrentRecord = .AbsolutePosition
txtSequencyNo = ![AssessmentSequence]
LoadDataToControls rstILAssessment, cboStrategiesLearning, False
LoadDataToControls rstILAssessment, cboStaffs, False
LoadDataToControls rstILAssessment, txtilassessment, True
LoadDataToControls rstILAssessment, chkGoals, True
LoadDataToControls rstILAssessment, cboExternalServices, True
LoadDataToControls rstILAssessment, cboAgencyServices, True
LoadDataToControls rstILAssessment, cboEquipment, True
LoadDataToControls rstILAssessment, cboServiceLevel, True
LoadDataToControls rstILAssessment, cboObserveResponse, True
LoadDataToControls rstILAssessment, cboRating, True
LoadDataToControls rstILAssessment, cboDifficulty, True
LoadDataToControls rstILAssessment, cboTimeFrame, False
LoadDataToControls rstILAssessment, cboFactors, True

If rst.State = adStateOpen Then rst.Close
rst.Open "Select * From IAssessment1 Where ConsumerID = " & CustomerID & " And Agency = "
    AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & _
    " And InterventionID = '013' And AssessmentSequence = " & ![AssessmentSequence] & _
    " Order By EntryDate", cnn, adOpenKeyset, adLockOptimistic

LoadDataToControls rst, cboYesNo, True
LoadDataToControls rst, cboAgree, True
rst.Close
End With

txtCurrentRecord.SetFocus
End Sub

Private Sub cmdPrint_Click()
Dim i As Integer

If Printers.Count = 0 Then
    MsgBox "No printer found associating with this computer. Please add a printer to it and th
ou can print out the Service agreement! ", _
        vbInformation + vbOKOnly, "Printer"
Else
    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select * from IAgreement Where AssessmentSequence = " & txtSequencyNo, cnn, ad
eyset, adLockOptimistic
    If Not rst.EOF Then ' if there an existing report, delete it then create a new report
        rst.Delete
    End If
    rst.Close
    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select * from IAgreement Where AssessmentSequence = " & txtSequencyNo, cnn, ad
eyset, adLockOptimistic
    With rst

```



```

If .BOF And .EOF Then
    .AddNew
    ![(CaseNumber)] = txtCaseNumber
    ![(Episode)] = txtEpisode
    ![(AssessmentSequence)] = txtSequencyNo
    ![(ConsumerName)] = txtConsumerName
    ![(AssessmentDate)] = txtilassessment(0)
    ![(AssessorName)] = cboStaffs.Text
    For i = 0 To chkGoals.Count - 1
        If chkGoals(i).Value = 0 Then
            .Fields(chkGoals(i).Tag) = "No"
        Else
            .Fields(chkGoals(i).Tag) = "Yes"
        End If
    Next
    For i = 0 To 7
        .Fields(cboYesNo(i).Tag) = cboYesNo(i).Text
    Next
    For i = 0 To cboEquipment.Count - 1
        .Fields(cboEquipment(i).Tag) = cboEquipment(i).Text
    Next
    For i = 0 To cboAgencyServices.Count - 1
        .Fields(cboAgencyServices(i).Tag) = cboAgencyServices(i).Text
    Next
    For i = 0 To cboExternalServices.Count - 1
        .Fields(cboExternalServices(i).Tag) = cboExternalServices(i).Text
    Next
    .Fields("TimeFrame") = cboTimeFrame.Text
    .Update
End If
End With
'rst.Requery
Set ILAgreement.DataSource = rst
ILAgreement.PrintReport
End If
End Sub

Private Sub cmdSave_Click()
    Dim i As Integer
    Dim colILAssessment As New Collection

    On Error GoTo errSave

    colILAssessment.Add cboRating
    colILAssessment.Add txtilassessment
    colILAssessment.Add cboStrategiesLearning
    colILAssessment.Add cboStaffs
    colILAssessment.Add cboLargerLVDeviceSize
    colILAssessment.Add cboEnlargerName
    colILAssessment.Add cboReaderName
    colILAssessment.Add cboBrailleLevel
    colILAssessment.Add chkGoals
    colILAssessment.Add cboExternalServices
    colILAssessment.Add cboAgencyServices
    colILAssessment.Add cboEquipment
    colILAssessment.Add cboServiceLevel
    colILAssessment.Add cboFactors
    colILAssessment.Add cboObserveResponse
    colILAssessment.Add cboDifficulty
    colILAssessment.Add cboTimeFrame

    If IsDate(txtilassessment(0)) Then ' If filing the assessment make sure the learn strategies
        ' and the factors are there

    If cboStrategiesLearning.ListIndex = -1 Then

```

```

        tabILAssessment.Tab = 9
        cboStrategiesLearning.SetFocus
        MsgBox "You had to select one of items from Strategy Learning!", vbInformation + vbOKOnly, "Strategy Learning"
        Exit Sub
    End If

    For i = 0 To cboFactors.Count - 1
        If cboFactors(i).ListIndex = -1 Then
            tabILAssessment.Tab = 9
            cboFactors(i).SetFocus
            MsgBox "You had to select one of items from Factors!", vbInformation + vbOKOnly, "Factors"
        End If
    Next

End If

If rstILAssessment.State = adStateOpen Then rstILAssessment.Close
rstILAssessment.Open "Select * From IAssessment Where ConsumerID = " & CustomerID & _
    " And Agency = " & AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & _
    " And InterventionID = '013' Order By AssessmentSequence DESC", cnn, _
    adOpenKeyset, adLockOptimistic

If SaveDataToDB(frmILAssessment, rstILAssessment, colILAssessment, "013", False) Then
    all public function to save data to DB

    ILSequenceNo = rstILAssessment![AssessmentSequence]
    rstILAssessment.Close
    For i = 1 To colILAssessment.Count
        colILAssessment.Remove 1
    Next

    rstILAssessment.Open "Select * From IAssessment1 Where ConsumerID = " & CustomerID & _
        " And Agency = " & AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & _
        " And InterventionID = '013' And AssessmentSequence = " & ILSequenceNo & _
        " Order By AssessmentSequence DESC", cnn, adOpenKeyset, adLockOptimistic

    colILAssessment.Add cboYesNo
    colILAssessment.Add cboAgree
    If SaveDataToDB(frmILAssessment, rstILAssessment, colILAssessment, "013", True) Then

        rstILAssessment.Close
        rstILAssessment.Open "Select * From IAssessment Where ConsumerID = " & CustomerID & _
            " And Agency = " & AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & _
            " And InterventionID = '013' Order By AssessmentSequence DESC", cnn, _
            adOpenKeyset, adLockOptimistic

        MsgBox "Successfully saved the Assessment information!", vbOKOnly, "Save"

        If IsDate(txtilassessment(0)) Then
            cnn.Execute "Update Interventions Set PreConsumerScore = " & txtConsumerScore & _
                " , PreProviderScore = " & txtProviderScore & " Where ConsumerID = " & CustomerID & _
                " And Agency = " & AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & _
                " And InterventionID = '013' And (EndDate = #12:00:00 PM# or EndDate IS NULL)"

            cnn.Execute "UPDATE IAssessment AS a INNER JOIN Interventions AS b ON (a.CaseNumber = b.CaseNumber) & _
                " AND (a.Episode = b.Episode) AND (a.ConsumerID = b.ConsumerID) SET a.InterventionSequence = b.InterventionSequence" & _
                " WHERE b.InterventionID = '013' and b.CaseNumber = " & CaseNumber & " AND (b.EndDate = #12:00:00PM# OR b.EndDate IS NULL)" & _
                " AND a.InterventionSequence = 0"

        End If
    End If

```

```

With rstILAssessment
    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then
            cmdPreviousRecord.Enabled = False
        Else
            cmdPreviousRecord.Enabled = True
        End If

        If .RecordCount = .AbsolutePosition Then
            cmdNextRecord.Enabled = False
        Else
            cmdNextRecord.Enabled = True
        End If
    End If
End With

If IsNewRecord Then
    rstILAssessment.Requery
    txtSequencyNo = rstILAssessment![AssessmentSequence]
End If

IsNewRecord = False
Set colILAssessment = Nothing
End If
End If
Exit Sub

errSave:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub

Private Sub Form_Activate()
On Error GoTo Errhandler

If IsVisualInfoClicked Then Exit Sub

txtConsumerName = FirstName & " " & LastName
txtPhone = PhoneNo
txtAddress = Address
txtCaseNumber = CaseNumber
txtEpisode = Episode
txtIntervention = "IL Assessment"

Set rst = New ADODB.Recordset 'create recordset object

LoadControlsItems cboStaffs, False 'Add every description from database to combobox or
tbox

Set rstILAssessment = New ADODB.Recordset
rstILAssessment.CursorLocation = adUseClient
rstILAssessment.Open "Select * From IAssessment Where ConsumerID = " & CustomerID & _
    " And Agency = " & AgencyCode & " And Episode = " & Episode & " And CaseNumber = " &
    CaseNumber & " And InterventionID = '013' Order By AssessmentSequence DESC", cnn, _
    adOpenKeyset, adLockOptimistic

With rstILAssessment
    txtRecordNo = .RecordCount
    If Not (.EOF And .BOF) Then 'Have record for this client and load it to
m
        txtCurrentRecord = .AbsolutePosition
        txtSequencyNo = ![AssessmentSequence]
        LoadDataToControls rstILAssessment, cboStrategiesLearning, False
        LoadDataToControls rstILAssessment, cboStaffs, False
        LoadDataToControls rstILAssessment, txtilassessment, True
        LoadDataToControls rstILAssessment, chkGoals, True
        LoadDataToControls rstILAssessment, cboExternalServices, True
        LoadDataToControls rstILAssessment, cboAgencyServices, True
    End If
End With

```

```

LoadDataToControls rstILAssessment, cboEquipment, True
LoadDataToControls rstILAssessment, cboServiceLevel, True
LoadDataToControls rstILAssessment, cboObserveResponse, True
LoadDataToControls rstILAssessment, cboRating, True
LoadDataToControls rstILAssessment, cboDifficulty, True
LoadDataToControls rstILAssessment, cboLargerLVDeviceSize, True
LoadDataToControls rstILAssessment, cboBrailleLevel, True
LoadDataToControls rstILAssessment, cboEnlargerName, False
LoadDataToControls rstILAssessment, cboReaderName, False
LoadDataToControls rstILAssessment, cboTimeFrame, False
LoadDataToControls rstILAssessment, cboFactors, True

rst.Open "Select * From IAssessment1 Where ConsumerID = " & CustomerID & " And Agency = " & _
AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & _
" And InterventionID = '013' And AssessmentSequence = " & ![AssessmentSequence] & _
" Order By AssessmentSequence", cnn, adOpenKeyset, adLockOptimistic
With rst
    If Not (.BOF And .EOF) Then
        LoadDataToControls rst, cboYesNo, True
        LoadDataToControls rst, cboAgree, True
    End If
End With

If .RecordCount <= 1 Then
    cmdNextRecord.Enabled = False
    cmdPreviousRecord.Enabled = False
ElseIf .RecordCount >= 2 Then
    cmdPreviousRecord.Enabled = False
    cmdNextRecord.Enabled = True
End If

If ![FinishDate] <> "12:00:00 PM" And IsDate(![FinishDate]) Then
    cmdSave.Enabled = False
    cmdPrint.Enabled = True
    cmdClear.Enabled = True
End If
Else
    cmdNextRecord.Enabled = False
    cmdPreviousRecord.Enabled = False
    txtCurrentRecord = 0
End If
End With

tabILAssessment.Tab = 0
txtConsumerName.SetFocus
Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"

End Sub

Private Sub Form_Unload(Cancel As Integer)
Dim response As Integer

If IsFileDateChecked Then
    If rst.State = adStateOpen Then rst.Close
    If rstILAssessment.State = adStateOpen Then rstILAssessment.Close
    Set rst = Nothing
    Set rstILAssessment = Nothing
    tabILAssessment.Tab = 0
    Set frmILAssessment = Nothing
Else
    If Trim(txtilassessment(0)) = "" And cmdSave.Enabled Then
        response = MsgBox("Have you finished IL Assessment ?", vbQuestion + vbYesNo, "File Date")
        If response = vbYes Then
            MsgBox "Please enter the File Date !", vbInformation + vbOKOnly, "File Date"
            txtilassessment(0).SetFocus
            Cancel = True
        End If
    End If
End If

```

```

Else
    If rst.State = adStateOpen Then rst.Close
    If rstILAssessment.State = adStateOpen Then rstILAssessment.Close
    Set rst = Nothing
    Set rstILAssessment = Nothing
    tabILAssessment.Tab = 0
    Set frmILAssessment = Nothing
End If
ElseIf IsDate(Trim(txtILAssessment(0))) And cmdSave.Enabled Then
    response = MsgBox("Are you sure you finished the IL Assessment?", vbQuestion + vbYesNo,
le Date")
    If response = vbYes Then
        MsgBox "Please click Save button to save the assessment information", vbInformation +
OKOnly, "File Date"
        cmdSave.SetFocus
        ElseIf cmdSave.Enabled = True Then
            txtILAssessment(0) = ""
        End If
    ElseIf IsDate(Trim(txtILAssessment(0))) And Not cmdSave.Enabled Then
        If rst.State = adStateOpen Then rst.Close
        If rstILAssessment.State = adStateOpen Then rstILAssessment.Close
        Set rst = Nothing
        Set rstILAssessment = Nothing
        tabILAssessment.Tab = 0
        Set frmILAssessment = Nothing
    End If
End If
End Sub

Private Sub CountGoals()
    Dim i As Integer
    For i = 0 To cboAgree.Count - 1
        Select Case i
            Case 0 To 19
                If cboAgree(i).Text = "Yes" Then chkGoals(0) = 1
            Case 20 To 42
                If cboAgree(i).Text = "Yes" Then chkGoals(1) = 1
            Case 43 To 50
                If cboAgree(i).Text = "Yes" Then chkGoals(2) = 1
            Case 51 To 60
                If cboAgree(i).Text = "Yes" Then chkGoals(3) = 1
            Case 61 To 73
                If cboAgree(i).Text = "Yes" Then chkGoals(4) = 1
        End Select
    Next
End Sub

Private Sub CountScore()
    Dim i As Integer
    Dim tmpValue As Integer
    Dim taskA As Single
    Dim taskB As Single
    Dim taskD As Single
    Dim taskE As Single
    Dim avgScore As Single

    If cboStrategiesLearning.ListIndex = -1 Then
        tabILAssessment.Tab = 9
        cboStrategiesLearning.SetFocus
        MsgBox "You have to select one of items from Strategy for Learning!", vbInformation + vbOK
, "Strategy Learning"
        Exit Sub
    End If

    For i = 0 To cboFactors.Count - 1
        If cboFactors(i).ListIndex = -1 Then
            tabILAssessment.Tab = 9
            cboFactors(i).SetFocus
            MsgBox "You have to select one of items from Factors!", vbInformation + vbOKOnly, "Fact

```

```

Exit Sub
End If
Next

For i = 0 To cboDifficulty.Count - 1 'count the score for task A - self assessment items
    If cboDifficulty(i).ListIndex = -1 Then
        taskA = taskA + 4
    Else
        tmpValue = cboDifficulty(i).ItemData(cboDifficulty(i).ListIndex)
        If tmpValue = 5 Then 'User select "Refused", count 0 for this item
            taskA = taskA + 0
        Else
            taskA = taskA + tmpValue
        End If
    End If
Next

tmpValue = 0
For i = 0 To cboRating.Count - 1 'provider rated items
    If (i = 13 Or i = 14 Or i = 25 Or i = 70 Or i = 90 Or i = 91) Then GoTo Here ' Do not count these
    If (i >= 34 And i <= 46) Then GoTo Here ' provider ratings
    If cboRating(i).ListIndex = -1 Then 'user doesn't select one, count 10 for this item
        taskB = taskB + 10
    Else
        tmpValue = cboRating(i).ItemData(cboRating(i).ListIndex)
        If tmpValue = 11 Then 'User select "N/A", count 10 for this item
            taskB = taskB + 10
        Else
            taskB = taskB + tmpValue
        End If
    End If
End If

Here:
Next

taskD = cboStrategiesLearning.ItemData(cboStrategiesLearning.ListIndex)

tmpValue = 0
For i = 0 To cboFactors.Count - 1
    Select Case cboFactors(i).ItemData(cboFactors(i).ListIndex)
        Case -3: tmpValue = 120
        Case -2: tmpValue = 110
        Case -1: tmpValue = 105
        Case 0: tmpValue = 91
        Case 1: tmpValue = 80
        Case 2: tmpValue = 60
        Case 3: tmpValue = 25
    End Select
    taskE = taskE + tmpValue
Next

txtConsumerScore = taskA * 2.5 ' Self Assessment Score
txtProviderScore = taskB * 0.1 ' Provider Score
avgScore = ((taskA * 2.5) + (taskB * 0.1) + taskD + (taskE / 6)) / 4

txtilassessment(2) = CLng(avgScore)
If avgScore > 100 Then avgScore = 100
txtilassessment(10) = Format(taskA * 2.5, "###")
txtilassessment(11) = Format((taskB * 100) / 740, "###")
txtilassessment(12) = Format(taskD, "###")
txtilassessment(13) = Format(taskE / 6, "###")
txtilassessment(14) = Format(avgScore, "###")

Select Case CLng(avgScore)
    Case 0 To 60: txtilassessment(1) = "Level 3"
    Case 61 To 80: txtilassessment(1) = "Level 2"
    Case 81 To 90: txtilassessment(1) = "Level 1"

```

```
Case 91 To 100:   txtILassessment(1) = "No Service Needed"
End Select
End Sub
```

```
Private Function getTabCaption(tabIndex As Integer) As String
```

```
Select Case tabIndex
Case 0
    getTabCaption = "Media for Reading/Writing 1"
Case 1
    getTabCaption = "Media for Reading/Writing 2"
Case 2
    getTabCaption = "Organization & Differentiating Skill 1"
Case 3
    getTabCaption = "Organization & Differentiating Skill 2"
Case 4
    getTabCaption = "Measurement"
Case 5
    getTabCaption = "Scanning Skills"
Case 6
    getTabCaption = "Safety & Cooking"
Case 7
    getTabCaption = "Observe And Inform Consumer 1"
Case 8
    getTabCaption = "Observe And Inform Consumer 2"
Case 9
    getTabCaption = "Learning Strategy and Factors"
Case 10
    getTabCaption = "IL Services Agreement"
End Select
```

```
End Function
```

```
Private Sub tabILAssessment_Click(PreviousTab As Integer)
Dim i As Integer
```

```
If PreviousTab = 9 Then
    For i = 0 To chkGoals.Count - 1
        chkGoals(i).Value = 0
    Next

    If cboServiceLevel(0).ListIndex = -1 Then
        cboTimeFrame.ListIndex = -1
    Else
        cboTimeFrame.ListIndex = cboServiceLevel(0).ListIndex - 1
    End If
End If
```

```
End If
```

```
If tabILAssessment.Tab = 0 Then
    frmILAssessment.Caption = "Media for Reading/Writing"
    cboStaffs.SetFocus
ElseIf tabILAssessment.Tab = 1 Then
    frmILAssessment.Caption = "Media for Reading/Writing"
    cboRating(87).SetFocus
ElseIf tabILAssessment.Tab = 2 Then
    frmILAssessment.Caption = "Organization & Differentiating Skill"
    cboDifficulty(7).SetFocus
ElseIf tabILAssessment.Tab = 3 Then
    frmILAssessment.Caption = "Organization & Differentiating Skill"
    cboRating(57).SetFocus
ElseIf tabILAssessment.Tab = 4 Then
    frmILAssessment.Caption = "Measurement"
    cboDifficulty(6).SetFocus
ElseIf tabILAssessment.Tab = 5 Then
    frmILAssessment.Caption = "Scanning Skills"
    cboDifficulty(3).SetFocus
ElseIf tabILAssessment.Tab = 6 Then
    frmILAssessment.Caption = "Safety & Cooking"
```

```
cboDifficulty(1).SetFocus
ElseIf tabILAssessment.Tab = 7 Then
    frmILAssessment.Caption = "Observe And Inform Consumer"
    cmdInstruction(2).SetFocus
ElseIf tabILAssessment.Tab = 8 Then
    frmILAssessment.Caption = "Observe And Inform Consumer"
    cboYesNo(17).SetFocus
ElseIf tabILAssessment.Tab = 9 Then
    frmILAssessment.Caption = "Learning Strategy and Factors"
    cboStrategiesLearning.SetFocus
ElseIf tabILAssessment.Tab = 10 Then
    CountGoals
    If Not IsGoalsClicked Then
        frmILAssessment.Caption = "IL Services Agreement"
        cmdInstruction(4).SetFocus
    End If
End If
End If
End Sub

Private Sub txtCurrentRecord_LostFocus()

If CheckForTab Then
    cboStaffs.SetFocus
    frmILAssessment.Caption = "Self Assessment"
End If
End Sub

Private Sub txtilassessment_Validate(Index As Integer, Cancel As Boolean)

If Index = 0 Or Index = 7 Or Index = 8 Then
    If Trim(txtilassessment(Index)) <> "" Then
        If Not IsDate(Trim(txtilassessment(Index))) Then
            MsgBox "Please enter date into this field !", vbInformation + vbOKOnly, "Date"
            txtilassessment(Index) = ""
            Cancel = True
        End If
    End If
End If
End If
End Sub
```



```
Option Explicit
Private tmpPosition As Long
Private rst As ADODB.Recordset
Private rstInterventions As ADODB.Recordset
Private rsScores As ADODB.Recordset
Private rsScores2 As ADODB.Recordset
Public SequenceNo As Long
Public InterventionCode As String
Private PreSelfScore As Integer
Private PreProviderScore As Integer
Private PreServiceScore As Single
Private PostServiceScore As Single
Private Functional As Double

Private Sub cmdClose_Click()
Unload Me
End Sub

Private Sub cmdEndOfService_Click()
frmEndOfService.Show vbModal
End Sub

Private Sub cmdNext_Click()
Dim i As Integer

If Trim(txtEndDate) = "" Then SaveInfoToDB

With rstInterventions
.MoveNext
If Not (.EOF) Then
txtCounter = .AbsolutePosition
InterventionCode = ![InterventionID]
txtStartDate = ![StartDate]

If ![EndDate] = "12:00:00 PM" Or Trim(![EndDate] & vbNullString) = "" Then
txtEndDate = ""
cmdSave.Enabled = True
Else
txtEndDate = ![EndDate]
cmdSave.Enabled = False
End If

LoadDataToControls rstInterventions, cboStaffs, False
LoadDataToControls rstInterventions, cboInterventionOutcome, False
LoadDataToControls rstInterventions, txtPreserviceScore, False
LoadDataToControls rstInterventions, txtPostServiceScore, False
LoadDataToControls rstInterventions, txtFunctionalChange, False

If Trim(![OutcomeNarrative] & vbNullString) = "" Or ![OutcomeNarrative] = Null Then
txtOutcomeNarrative = ""
Else
txtOutcomeNarrative = ![OutcomeNarrative]
End If

tmpPosition = .AbsolutePosition
SequenceNo = ![InterventionSequence]
InterventionCode = ![InterventionID]

If rst.State = adStateOpen Then rst.Close
rst.Open "Select Description From InterventionCodes Where Code = '" & InterventionCode
", -
cnn, adOpenForwardOnly, adLockReadOnly
txtIntervention = rst![Description]
txtIntervention.SetFocus

If .RecordCount > 1 Then
If .AbsolutePosition = 1 Then
cmdPrevious.Enabled = False
cmdNext.Enabled = True
```

```

ElseIf .AbsolutePosition < .RecordCount Then
    cmdPrevious.Enabled = True
    cmdNext.Enabled = True
ElseIf .RecordCount = .AbsolutePosition Then
    cmdPrevious.Enabled = True
    cmdNext.Enabled = False
End If
End If
End If
End With

End Sub

Private Sub cmdPrevious_Click()
Dim i As Integer

If Trim(txtEndDate) = "" Then SaveInfoToDB

With rstInterventions
.MovePrevious
If Not (.BOF) Then
    txtCounter = .AbsolutePosition
    InterventionCode = ![InterventionID]
    txtStartDate = ![StartDate]

    If ![EndDate] = "12:00:00 PM" Or Trim(![EndDate] & vbNullString) = "" Then
        txtEndDate = ""
        cmdSave.Enabled = True
    Else
        txtEndDate = ![EndDate]
        cmdSave.Enabled = False
    End If

    LoadDataToControls rstInterventions, cboStaffs, False
    LoadDataToControls rstInterventions, cboInterventionOutcome, False
    LoadDataToControls rstInterventions, txtPreserviceScore, False
    LoadDataToControls rstInterventions, txtPostServiceScore, False
    LoadDataToControls rstInterventions, txtFunctionalChange, False

    If Trim(![OutcomeNarrative] & vbNullString) = "" Or ![OutcomeNarrative] = Null Then
        txtOutcomeNarrative = ""
    Else
        txtOutcomeNarrative = ![OutcomeNarrative]
    End If

    SequenceNo = ![InterventionSequence]
    InterventionCode = ![InterventionID]

    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select Description From InterventionCodes Where Code = '" & InterventionCode &
    ", -
        cnn, adOpenForwardOnly, adLockReadOnly
    txtIntervention = rst![Description]

    txtIntervention.SetFocus
    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then
            cmdPrevious.Enabled = False
            cmdNext.Enabled = True
        ElseIf .AbsolutePosition < .RecordCount Then
            cmdPrevious.Enabled = True
            cmdNext.Enabled = True
        ElseIf .RecordCount = .AbsolutePosition Then
            cmdPrevious.Enabled = True
            cmdNext.Enabled = False
        End If
    End If
End If
/ End If
End With

```

End Sub

Private Sub cmdReferral\_Click()

frmReferral.Show vbModal  
End Sub

Private Sub cmdSave\_Click()

SaveInfoToDB  
'cmdNext.Enabled = True

If (IsDate(txtEndDate) = True) Then  
    CalculateScore  
End If

MsgBox "Successfully save the information!", vbOKOnly, "Save Info"  
End Sub

Private Sub cmdVisits\_Click()

'Dim rst As New ADODB.Recordset  
'Dim SQLStr As String

'If (InterventionCode = "014" Or InterventionCode = "013" Or InterventionCode = "010" Or InterventionCode = "050" Or InterventionCode = "160") Then

    'Select Case InterventionCode

        'Case "010"

            'SQLStr = "Select Agency From SocialAssessment Where ConsumerID=" & CustomerID & \_  
                        " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & \_  
                        "' And Episode = '" & CLng(Trim(txtEpisode))

        'Case "014"

            'SQLStr = "Select Agency From MobilityAssessments Where ConsumerID=" & CustomerID & \_  
                        " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & \_  
                        "' And Episode = '" & CLng(Trim(txtEpisode))

        'Case "013"

            'SQLStr = "Select Agency From IAssessment Where ConsumerID=" & CustomerID & \_  
                        " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & \_  
                        "' And Episode = '" & CLng(Trim(txtEpisode))

        'Case "050"

            'SQLStr = "Select Agency From EmploymentAssessment Where ConsumerID=" & CustomerID & \_  
                        " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & \_  
                        "' And Episode = '" & CLng(Trim(txtEpisode))

        'Case "160"

            'SQLStr = "Select Agency From ComputerAssessment Where ConsumerID=" & CustomerID & \_  
                        " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & \_  
                        "' And Episode = '" & CLng(Trim(txtEpisode))

    'End Select

    'With rst

        '.Open SQLStr, cnn, adOpenForwardOnly, adLockReadOnly

        'If .BOF And .EOF Then

            'MsgBox "Please create an Assessment for the selected intervention before making visits records!", -

            vbOKOnly + vbInformation, "Visits"

        'Else

            'frmVisits1.Show vbModal

        'End If

    'End With

    'rst.Close

    'Set rst = Nothing

    'Else

        'frmVisits1.Show vbModal

    'End If

End Sub

Private Sub Form\_Load()

Dim i As Integer

On Error GoTo Errhandler

```
LoadControlsItems cboStaffs, False
LoadControlsItems cboInterventionOutcome, False
```

```
txtFirstName = FirstName
txtLastName = LastName
txtCaseNumber = CaseNumber
txtEpisode = Episode
```

```
Set rst = New ADODB.Recordset
rst.CursorLocation = adUseClient
Set rstInterventions = New ADODB.Recordset
rstInterventions.CursorLocation = adUseClient
rstInterventions.Open "Select InterventionID, Staffname, StartDate, EndDate, InterventionSeque
," & _
    " InterventionOutcome, OutcomeNarrative, PreServiceScore, PostServiceScore, FunctionalCha
From Interventions Where ConsumerID=" & _
    CustomerID & " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' And
sode = '" & _
    CLng(Trim(txtEpisode)) & " Order By InterventionSequence DESC", cnn, adOpenDynamic, adLockO
mistic
```

```
With rstInterventions
```

```
    If Not (.BOF And .EOF) Then
        txtCounter = .AbsolutePosition
        InterventionCode = ![InterventionID]
        txtStartDate = ![StartDate]
        LoadDataToControls rstInterventions, cboStaffs, False,
        LoadDataToControls rstInterventions, cboInterventionOutcome, False
        LoadDataToControls rstInterventions, txtPreserviceScore, False
        LoadDataToControls rstInterventions, txtPostServiceScore, False
        LoadDataToControls rstInterventions, txtFunctionalChange, False
```

```
    If ![EndDate] = "12:00:00 PM" Or Trim(![EndDate] & vbNullString) = "" Then
        txtEndDate = ""
        cmdSave.Enabled = True
    Else
        txtEndDate = ![EndDate]
        cmdSave.Enabled = False
    End If
```

```
    If Trim(![OutcomeNarrative] & vbNullString) = "" Or ![OutcomeNarrative] = Null Then
        txtOutcomeNarrative = ""
    Else
        txtOutcomeNarrative = ![OutcomeNarrative]
    End If
```

```
    tmpPosition = .AbsolutePosition
    SequenceNo = ![InterventionSequence]
    InterventionCode = ![InterventionID]
```

```
    txtPostProviderScore = ![postProviderscore] & vbNullString
    txtPostConsumerScore = ![postConsumerscore] & vbNullString
```

```
rst.Open "Select Description From InterventionCodes Where Code = '" & InterventionCode
," & _
    cnn, adOpenForwardOnly, adLockReadOnly
txtIntervention = rst![Description]
```

```
If .RecordCount > 1 Then
    If .AbsolutePosition = 1 Then
        cmdPrevious.Enabled = False
        cmdNext.Enabled = True
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPrevious.Enabled = True
        cmdNext.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPrevious.Enabled = True
        cmdNext.Enabled = False
```

```

        End If
    End If
End With

If rst.State = adStateOpen Then rst.Close.
txtTotal = rstInterventions.RecordCount
Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbInformation, "Form Load"
End Sub

Private Sub Form_Unload(Cancel As Integer)
'Dim response As Integer

'response = MsgBox("Would you like save the Intervention information ?", vbQuestion + vbYesNo,
lose")
'If response = vbYes Then
'    MsgBox "Please use Save button to save it!", vbInformation + vbOKOnly, "Close"
'    Cancel = True
'Else
If rst.State = adStateOpen Then rst.Close
rstInterventions.Close
Set rst = Nothing
Set rstInterventions = Nothing
Set frmIntervention1 = Nothing
'End If
End Sub

Private Sub txtEndDate_Validate(Cancel As Boolean)

If Trim(txtEndDate) = "" Then
    Cancel = False
ElseIf Not IsDate(txtEndDate) Then
    MsgBox "Please enter date in this field!", vbInformation + vbOKOnly, "Date Validate"
    Cancel = True
End If
End Sub

Private Sub SaveInfoToDB()
Dim tmpDate As Date
Dim tmpStr As String
Dim tmpstr1 As String
Dim tmpstr2 As String

If cboStaffs.ListIndex = -1 Then
    tmpStr = ""
Else
    tmpStr = cboStaffs.ItemData(cboStaffs.ListIndex)
End If

If cboInterventionOutcome.ListIndex = -1 Then
    tmpstr1 = ""
Else
    tmpstr1 = cboInterventionOutcome.ItemData(cboInterventionOutcome.ListIndex)
End If

If Trim(txtEndDate) = "" Then
    tmpDate = "12:00:00 PM"
Else
    tmpDate = txtEndDate.Text
End If

cnn.Execute "Update Interventions Set StaffName = '" & tmpStr & "', EndDate = '" & tmpDate &
"', InterventionOutcome = '" & tmpstr1 & "', OutcomeNarrative = '" & _
txtOutcomeNarrative.Text & "' Where InterventionSequence = '" & SequenceNo
tmpPosition = rstInterventions.AbsolutePosition
rstInterventions.Requery

```

```
rstInterventions.AbsolutePosition = tmpPosition
End Sub
```

```
Private Sub CalculateScore() ' Calculate the Intervention Pre/Post Scores
```

```
    If InterventionCode = "013" Then ' IL
        Calculate_IL_Score
    ElseIf InterventionCode = "014" Then ' Mobility
        Calculate_OM_Score
    ElseIf InterventionCode = "050" Then ' Employment
        Calculate_EM_Score
    ElseIf InterventionCode = "010" Then ' Social Service
        Calculate_SS_Score
    ElseIf InterventionCode = "160" Then ' Computer Skills
        Calculate_CS_Score
    End If
```

```
End Sub
```

```
Private Sub Calculate_IL_Score()
```

```
    Dim TotalObjectives As Integer
    Dim Objectives As Integer
    Dim ObjAchieve As Long
    Dim AssessSeq As Integer
    Dim ProgressSeq As Integer
    Dim GoalSeq As Integer
    Dim Temp As Single
    Dim Functional As Single
    Dim Counter As Integer
```

```
    PreSelfScore = 0
    PreProviderScore = 0
    AssessSeq = 0
    ProgressSeq = 0
    ObjAchieve = 0
    Temp = 0
    Counter = 0
    TotalObjectives = 74
```

```
    ' Calculate score only if there are visit records for the intervention
```

```
    Set rsScores = New ADODB.Recordset
    rsScores.CursorLocation = adUseClient
```

```
    rsScores.Open "Select Duration From Visits Where ConsumerID = " & CustomerID & _
        " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' & _
        " And Episode = '" & Episode & "' And InterventionSequence = '" & SequenceNo & "' & _
        " And (Type = '5' or Type = '10')", cnn, adOpenForwardOnly, adLockReadOnly
```

```
    With rsScores
```

```
        If Not (.BOF And .EOF) Then
```

```
            Temp = CInt(![Duration])
```

```
            If Temp = 0 Then ' 0 minutes skip scoring
```

```
                Exit Sub
```

```
            End If
```

```
        Else
```

```
            Exit Sub
```

```
        ' No visit records skip scoring
```

```
        End If
```

```
    End With
```

```
    rsScores.Close
```

```
    Set rsScores = Nothing
```

```
    ' PreService Score
```

```
    Set rsScores = New ADODB.Recordset
```

```
    rsScores.CursorLocation = adUseClient
```

```
    rsScores.Open "Select TaskB, AssessmentSequence From ILAssessment Where ConsumerID = " & Cust  
ID & _
```

```
        " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' & _
```

```
        " And Episode = '" & Episode & "' And InterventionSequence = '" & SequenceNo, _
```

```
        cnn, adOpenForwardOnly, adLockReadOnly
```

```

With rsScores
  If Not (.BOF And .EOF) Then
    PreProviderScore = ![taskB]
    AssessSeq = ![AssessmentSequence]
  Else
    Exit Sub
  End If
End With

rsScores.Close
Set rsScores = Nothing

PreServiceScore = PreProviderScore

' PostService Score
Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select ProgressSeq From ProgressReport Where ConsumerID = " & CustomerID & _
  " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' & _
  " And Episode = '" & Episode & "' And InterventionID = '013'" & _
  " And AssessmentSequence = " & AssessSeq, cnn, adOpenForwardOnly, adLockReadOr

With rsScores
  If Not (.BOF And .EOF) Then
    ProgressSeq = ![ProgressSeq]
  Else
    Exit Sub
  End If
End With

rsScores.Close
Set rsScores = Nothing

' Get the GoalSeq for all goals
Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select GoalSeq From GoalAchieve Where ConsumerID = " & CustomerID & _
  " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' & _
  " And Episode = '" & Episode & "' And InterventionID = '013'" & _
  " And ProgressSeq = " & ProgressSeq, cnn, adOpenForwardOnly, adLockReadOnly

With rsScores
  If Not (.BOF And .EOF) Then
    While Not .EOF
      GoalSeq = ![GoalSeq]
      ' Get the achievement rating for all objectives
      Set rsScores2 = New ADODB.Recordset
      rsScores2.CursorLocation = adUseClient
      rsScores2.Open "Select Achievement From ObjectiveAchieve Where ConsumerID = " & Cust
rID & _
        " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' & _
        " And Episode = '" & Episode & "' And InterventionID = '013'" & _
        " And GoalSeq = " & GoalSeq, cnn, adOpenForwardOnly, adLockReadOnly

      Objectives = Objectives + rsScores2.RecordCount

    With rsScores2
      If Not (.BOF And .EOF) Then
        While Not .EOF
          Temp = ![Achievement]
          ObjAchieve = ObjAchieve + Temp
          .MoveNext
        Wend
      Else
        rsScores2.Close
        Set rsScores2 = Nothing
        rsScores.Close
        Set rsScores = Nothing
        Exit Sub
      End If
    End With
  End With
End With

```

```

        .MoveNext
    Wend
Else
    Exit Sub ' No goals
End If
End With

rsScores2.Close
Set rsScores2 = Nothing
rsScores.Close
Set rsScores = Nothing

Temp = TotalObjectives - Objectives
If (Temp > 0) Then
    For Counter = 1 To Temp
        ObjAchieve = ObjAchieve + 10 ' Goals that were not rated are scored a 10
    Next
End If

PostServiceScore = ((ObjAchieve * 100) / 740)
Temp = PostServiceScore - PreServiceScore
Functional = Format$(((Temp / PreServiceScore) * 100), "Fixed")

txtPreserviceScore.Text = PreServiceScore
txtPostServiceScore.Text = PostServiceScore
txtFunctionalChange.Text = Functional

cnn.Execute "Update Interventions Set PreServiceScore = " & PreServiceScore & _
    ", PostServiceScore = " & PostServiceScore & ", FunctionalChange = " & Functional & _
    " Where ConsumerID = " & CustomerID & _
    " And Agency = " & AgencyCode & " And CaseNumber = " & CaseNumber & " & _
    " And Episode = " & Episode & " And InterventionSequence = " & SequenceNo

tmpPosition = rstInterventions.AbsolutePosition
rstInterventions.Requery
rstInterventions.AbsolutePosition = tmpPosition

End Sub

Private Sub Calculate_OM_Score()

Dim TotalObjectives As Integer
Dim Objectives As Integer
Dim ObjAchieve As Long
Dim AssessSeq As Integer
Dim GoalSeq As Integer
Dim ProgressSeq As Integer
Dim Temp As Single
Dim Functional As Single
Dim Counter As Integer

PreSelfScore = 0
PreProviderScore = 0
AssessSeq = 0
ProgressSeq = 0
ObjAchieve = 0
Temp = 0
Counter = 0
TotalObjectives = 44

' Calculate score only if there are visit records for the intervention
Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select Duration From Visits Where ConsumerID = " & CustomerID & _
    " And Agency = " & AgencyCode & " And CaseNumber = " & CaseNumber & " & _
    " And Episode = " & Episode & " And InterventionSequence = " & SequenceNo. & _
    " And (Type = '5' or Type = '10')", cnn, adOpenForwardOnly, adLockReadOnly

With rsScores

```



```

If Not (.BOF And .EOF) Then
    Temp = CInt(![Duration])
    If Temp = 0 Then      ' 0 minutes skip scoring
        Exit Sub
    End If
Else
    Exit Sub      ' No visit records skip scoring
End If
End With

rsScores.Close
Set rsScores = Nothing

' PreService Score

Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select TaskB, AssessmentSequence From MobilityAssessments Where ConsumerID = " & _
CustomerID & " " & AgencyCode & " " & CaseNumber & " " & CaseNumber & " " & _
" And Episode = " & Episode & " And InterventionSequence = " & SequenceNo, _
cnn, adOpenForwardOnly, adLockReadOnly

With rsScores
    If Not (.BOF And .EOF) Then      ' Get the preprovider ratings for
        PreProviderScore = ![taskB]    this intervention sequence
        AssessSeq = ![AssessmentSequence]
    Else
        Exit Sub      ' No assessments for the consumer skip scoring
    End If
End With

rsScores.Close
Set rsScores = Nothing

PreServiceScore = PreProviderScore

' PostService Score
Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select ProgressSeq From ProgressReport Where ConsumerID = " & CustomerID & " " & _
AgencyCode & " " & CaseNumber & " " & CaseNumber & " " & _
Episode & " And InterventionID = '014' " & _
AssessSeq, cnn, adOpenForwardOnly, adLockReadOnly

With rsScores
    If Not (.BOF And .EOF) Then      ' Get the Progress sequence for this intervention
        ProgressSeq = ![ProgressSeq]
    Else
        Exit Sub      ' No progress report skip scoring
    End If
End With

rsScores.Close
Set rsScores = Nothing

' Get the GoalSeq for all goals
Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select GoalSeq From GoalAchieve Where ConsumerID = " & CustomerID & " " & _
AgencyCode & " " & CaseNumber & " " & CaseNumber & " " & _
Episode & " And InterventionID = '013' " & _
ProgressSeq, cnn, adOpenForwardOnly, adLockReadOnly

With rsScores
    If Not (.BOF And .EOF) Then
        While Not .EOF
            GoalSeq = ![GoalSeq]
            ' Get the achievement rating for all objectives
            Set rsScores2 = New ADODB.Recordset
            rsScores2.CursorLocation = adUseClient

```

```

rsScores2.Open "Select Achievement From ObjectiveAchieve Where ConsumerID = " & Custom
rID & -
    " And Agency = " & AgencyCode & " And CaseNumber = " & CaseNumber & " & -
    " And Episode = " & Episode & " And InterventionID = '014'" & -
    " And GoalSeq = " & GoalSeq, cnn, adOpenForwardOnly, adLockReadOnly

Objectives = rsScores.RecordCount

With rsScores2
    If Not (.BOF And .EOF) Then
        While Not .EOF
            Temp = ![Achievement]
            ObjAchieve = ObjAchieve + Temp
            .MoveNext
        Wend
    Else
        rsScores2.Close
        Set rsScores2 = Nothing
        rsScores.Close
        Set rsScores = Nothing
        Exit Sub ' No achievements skip scoring
    End If
End With

Wend

Else
    Exit Sub ' No goals
End If

End With

rsScores.Close
Set rsScores = Nothing

Temp = TotalObjectives - Objectives
If (Temp > 0) Then
    For Counter = 1 To Temp
        ObjAchieve = ObjAchieve + 10 ' Goals that were not rated are scored a 10
    Next
End If

PostServiceScore = (ObjAchieve * 100) / 440
Temp = PostServiceScore - PreServiceScore
Functional = Format$(((Temp / PreServiceScore) * 100), "Fixed")

txtPreserviceScore.Text = PreServiceScore
txtPostServiceScore.Text = PostServiceScore
txtFunctionalChange.Text = Functional

cnn.Execute "Update Interventions Set PreServiceScore = " & PreServiceScore & -
    ", PostServiceScore = " & PostServiceScore & ", FunctionalChange = " & Functional & -
    " Where ConsumerID = " & CustomerID & -
    " And Agency = " & AgencyCode & " And CaseNumber = " & CaseNumber & " & -
    " And Episode = " & Episode & " And InterventionSequence = " & SequenceNo

tmpPosition = rstInterventions.AbsolutePosition
rstInterventions.Requery
rstInterventions.AbsolutePosition = tmpPosition

End Sub

Private Sub Calculate_EM_Score()

End Sub

Private Sub Calculate_SS_Score()

End Sub

Private Sub Calculate_CS_Score()

```

```

Dim TotalObjectives As Integer
Dim Objectives As Integer
Dim ObjAchieve As Long
Dim GoalSeq As Integer
Dim AssessSeq As Integer
Dim ProgressSeq As Integer
Dim Temp As Single
Dim Functional As Single
Dim Counter As Integer

```

```

PreSelfScore = 0
PreProviderScore = 0
AssessSeq = 0
ProgressSeq = 0
ObjAchieve = 0
Temp = 0
Counter = 0
TotalObjectives = 7

```

```

' Calculate score only if there are visit records for the intervention

```

```

Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select Duration From Visits Where ConsumerID = " & CustomerID & _
    " And Agency = " & AgencyCode & " And CaseNumber = " & CaseNumber & " " & _
    " And Episode = " & Episode & " And InterventionSequence = " & SequenceNo & _
    " And (Type = '5' or Type = '10')", cnn, adOpenForwardOnly, adLockReadOnly

```

```

With rsScores
    If Not (.BOF And .EOF) Then
        Temp = CInt(![Duration])
        If Temp = 0 Then ' 0 minutes skip scoring
            Exit Sub
        End If
    Else
        Exit Sub ' No visit records skip scoring
    End If
End With

```

```

rsScores.Close
Set rsScores = Nothing

```

```

' PreService Score

```

```

Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select TaskB, AssessmentSequence From ComputerAssessment Where ConsumerID = " & _
    CustomerID & _
    " And Agency = " & AgencyCode & " And CaseNumber = " & CaseNumber & " " & _
    " And Episode = " & Episode & " And InterventionSequence = " & SequenceNo, _
    cnn, adOpenForwardOnly, adLockReadOnly

```

```

With rsScores
    If Not (.BOF And .EOF) Then
        PreProviderScore = ![taskB] ' Get the preprovider ratings for
        AssessSeq = ![AssessmentSequence] ' this intervention sequence
    Else
        Exit Sub ' No assessments for the consumer skip scoring
    End If
End With

```

```

rsScores.Close
Set rsScores = Nothing

```

```

PreServiceScore = PreProviderScore

```

```

' PostService Score

```

```

Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select ProgressSeq From ProgressReport Where ConsumerID = " & CustomerID & _

```

```

    " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' & _
    " And Episode = '" & Episode & "' And InterventionID = '160'" & _
    " And AssessmentSequence = '" & AssessSeq, cnn, adOpenForwardOnly, adLockReadOn:

```

```

With rsScores
    If Not (.BOF And .EOF) Then
        ProgressSeq = ![ProgressSeq]
        ' Get the Progress sequence for this intervention
    Else
        Exit Sub
        ' No progress report skip scoring
    End If
End With

```

```

rsScores.Close
Set rsScores = Nothing

```

```

Set rsScores = New ADODB.Recordset
rsScores.CursorLocation = adUseClient
rsScores.Open "Select Distinct GoalID, Achievement From GoalAchieve Where ConsumerID = '" & ConsumerID & "'
    " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' & _
    " And Episode = '" & Episode & "' And InterventionID = '160'" & _
    " And ProgressSeq = '" & ProgressSeq, cnn, adOpenForwardOnly, adLockReadOnly

```

```

Objectives = rsScores.RecordCount

```

```

With rsScores
    If Not (.BOF And .EOF) Then
        ' Get the Achievement rating for each goal
        While Not .EOF
            Temp = ![Achievement]
            ObjAchieve = ObjAchieve + Temp
            .MoveNext
        Wend
    Else
        Exit Sub
        ' No achievements skip scoring
    End If
End With

```

```

rsScores.Close
Set rsScores = Nothing

```

```

Temp = TotalObjectives - Objectives
If (Temp > 0) Then
    For Counter = 1 To Temp
        ObjAchieve = ObjAchieve + 4
        ' Goals that were not rated are scored a 4
    Next
End If

```

```

PostServiceScore = ObjAchieve * 28
Temp = PostServiceScore - PreServiceScore
Functional = Format$((Temp / PreServiceScore) * 100), "Fixed")

```

```

txtPreserviceScore.Text = PreServiceScore
txtPostServiceScore.Text = PostServiceScore
txtFunctionalChange.Text = Functional

```

```

cnn.Execute "Update Interventions Set PreServiceScore = '" & PreServiceScore & "'
    ", PostServiceScore = '" & PostServiceScore & "', FunctionalChange = '" & Functional & "'
    " Where ConsumerID = '" & CustomerID & "'
    " And Agency = '" & AgencyCode & "' And CaseNumber = '" & CaseNumber & "' & _
    " And Episode = '" & Episode & "' And InterventionSequence = '" & SequenceNo

```

```

tmpPosition = rstInterventions.AbsolutePosition
rstInterventions.Requery
rstInterventions.AbsolutePosition = tmpPosition

```

```

End Sub

```

```
Private GoalSeq As Long
Private ObjectiveSeq As Long
Private rstIRA As ADODB.Recordset
Private rstGoal As ADODB.Recordset
Private rstObjectives As ADODB.Recordset

Private Sub cboStaffName_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboStaffName.ListIndex = -1
    End If
End Sub

Private Sub cboGSuccessful_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboGSuccessful.ListIndex = -1
    End If
End Sub

Private Sub cboOSuccessful_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboOSuccessful.ListIndex = -1
    End If
End Sub

Private Sub cmdClose_Click()
    Unload Me
End Sub

Private Sub cmdNextObjective_Click()

cmdPreviousObjective.Enabled = True
With rstObjectives
    .MoveNext
    If rstObjectives.RecordCount > 1 Then
        If rstObjectives.AbsolutePosition = 1 Then
            cmdPreviousObjective.Enabled = False
            cmdNextObjective.Enabled = True
        ElseIf rstObjectives.AbsolutePosition < .RecordCount Then
            cmdPreviousObjective.Enabled = True
            cmdNextObjective.Enabled = True
        ElseIf rstObjectives.RecordCount = .AbsolutePosition Then
            cmdPreviousObjective.Enabled = True
            cmdNextObjective.Enabled = False
        End If
    Else
        cmdNextObjective.Enabled = False
        cmdPreviousObjective.Enabled = False
    End If

    LoadDataToControls rstObjectives, cboOSuccessful, False
    LoadDataToControls rstObjectives, txtOCompletionDate, False
    txtObjectives = rstObjectives![Description]
    txtCurrentObjective = rstObjectives.AbsolutePosition
    ObjectiveSeq = ![ObjectiveSeq]
End With
End Sub

Private Sub cmdNextRecord_Click()

With rstIRA
    .MoveNext
    txtTotalRecord = .RecordCount
    txtCurrentRecord = .AbsolutePosition
    ProgressReportSeq ![ProgressSeq]
    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then
            cmdPreviousRecord.Enabled = False
            cmdNextRecord.Enabled = True
        ElseIf .AbsolutePosition < .RecordCount Then
            cmdPreviousRecord.Enabled = True
        End If
    End If
End With
```

```

        cmdNextRecord.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = False
    End If
End If
End With
End Sub

Private Sub cmdPreviousGoal_Click()
With rstGoal
    .MovePrevious
    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then
            cmdPreviousGoal.Enabled = False
            cmdNextGoal.Enabled = True
        ElseIf .AbsolutePosition < .RecordCount Then
            cmdPreviousGoal.Enabled = True
            cmdNextGoal.Enabled = True
        ElseIf .RecordCount = .AbsolutePosition Then
            cmdPreviousGoal.Enabled = True
            cmdNextGoal.Enabled = False
        End If
    Else
        cmdNextGoal.Enabled = False
        cmdPreviousGoal.Enabled = False
    End If

    LoadDataToControls rstGoal, cboGSuccessful, False
    LoadDataToControls rstGoal, txtGCompletionDate, False
    txtCurrentGoal = .AbsolutePosition
    txtGoal = rstGoal![Description]
    GoalSeq = rstGoal![GoalSeq]
    If rstObjectives.State = adStateOpen Then rstObjectives.Close
    rstObjectives.Open "Select OA.GoalSeq,O.Description,OA.Achievement,OA.CompletionDate,OA.ObjectiveID from " & _
        "ObjectiveAchieve OA, Objectives O where OA.GoalID = O.GoalID and OA.ObjectiveID = C" & _
        "ectiveID " & _
        "and OA.Episode = " & Episode & " And OA.InterventionID = " & frmIRAMenu.tmpIntervenID & _
        " And OA.ConsumerID = " & CustomerID & " And OA.GoalSeq = " & GoalSeq, cnn, adOpenStatic, adLockOptimistic
    If Not rstObjectives.EOF Then
        rstObjectives.MoveFirst
        If rstObjectives.RecordCount > 1 Then
            cmdNextObjective.Enabled = True
        Else
            cmdNextObjective.Enabled = False
        End If

        LoadDataToControls rstObjectives, cboOSuccessful, False
        LoadDataToControls rstObjectives, txtOCompletionDate, False
        txtObjectives = rstObjectives![Description]
        txtObjectiveCount = rstObjectives.RecordCount
        txtCurrentObjective = 1
        Set grdObjectives.DataSource = rstObjectives
        ObjectiveSeq = rstObjectives![ObjectiveSeq]
    Else
        cmdNextObjective.Enabled = False
        cmdPreviousObjective.Enabled = False
    End If
End With
End Sub

Private Sub cmdNextGoal_Click()
With rstGoal
    .MoveNext
    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then

```

```

        cmdPreviousGoal.Enabled = False
        cmdNextGoal.Enabled = True
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPreviousGoal.Enabled = True
        cmdNextGoal.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousGoal.Enabled = True
        cmdNextGoal.Enabled = False
    End If
Else
    cmdNextGoal.Enabled = False
    cmdPreviousGoal.Enabled = False
End If

LoadDataToControls rstGoal, cboGSuccessful, False
LoadDataToControls rstGoal, txtGCompletionDate, False
txtCurrentGoal = .AbsolutePosition
txtGoal = rstGoal![Description]
GoalSeq = rstGoal![GoalSeq]
If rstObjectives.State = adStateOpen Then rstObjectives.Close
rstObjectives.Open "Select OA.GoalSeq,O.Description,OA.Achievement,OA.CompletionDate, OA.ObjectiveSeq from " & _
    "ObjectiveAchieve OA, Objectives O where OA.GoalID = O.GoalID and OA.ObjectiveID = O.ObjectiveID " & _
    "and OA.Episode = " & Episode & " And OA.InterventionID = " & frmIRAMenu.tmpInterventionID & _
    " And OA.ConsumerID = " & CustomerID & " And OA.GoalSeq = " & rstGoal![GoalSeq], cmdOpenDynamic, adLockOptimistic
If Not rstObjectives.EOF Then
    rstObjectives.MoveFirst
    If rstObjectives.RecordCount > 1 Then
        cmdNextObjective.Enabled = True
    Else
        cmdNextObjective.Enabled = False
    End If
End If

LoadDataToControls rstObjectives, cboOSuccessful, False
LoadDataToControls rstObjectives, txtOCompletionDate, False
txtObjectives = rstObjectives![Description]
txtObjectiveCount = rstObjectives.RecordCount
txtCurrentObjective = rstObjectives.AbsolutePosition
Set grdObjectives.DataSource = rstObjectives
ObjectiveSeq = rstObjectives![ObjectiveSeq]
Else
    cmdNextObjective.Enabled = False
    cmdPreviousObjective.Enabled = False
End If
End With
End Sub

Private Sub cmdPreviousObjective_Click()

cmdNextObjective.Enabled = True
With rstObjectives
    .MovePrevious
    If rstObjectives.RecordCount > 1 Then
        If rstObjectives.AbsolutePosition = 1 Then
            cmdPreviousObjective.Enabled = False
            cmdNextObjective.Enabled = True
        ElseIf rstObjectives.AbsolutePosition < .RecordCount Then
            cmdPreviousObjective.Enabled = True
            cmdNextObjective.Enabled = True
        ElseIf rstObjectives.RecordCount = .AbsolutePosition Then
            cmdPreviousObjective.Enabled = True
            cmdNextObjective.Enabled = False
        End If
    Else
        cmdNextObjective.Enabled = False
        cmdPreviousObjective.Enabled = False
    End If
End If

```

```
LoadDataToControls rstObjectives, cboOSuccessful, False
LoadDataToControls rstObjectives, txtOCompletionDate, False
txtObjectives = rstObjectives![Description]
txtCurrentObjective = rstObjectives.AbsolutePosition
ObjectiveSeq = ![ObjectiveSeq]
End With
End Sub
```

```
Private Sub cmdPreviousRecord_Click()
```

```
With rstIRA
    .MovePrevious
    txtTotalRecord = .RecordCount
    txtCurrentRecord = .AbsolutePosition
    ProgressReportSeq ![ProgressSeq]
    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then
            cmdPreviousRecord.Enabled = False
            cmdNextRecord.Enabled = True
        ElseIf .AbsolutePosition < .RecordCount Then
            cmdPreviousRecord.Enabled = True
            cmdNextRecord.Enabled = True
        ElseIf .RecordCount = .AbsolutePosition Then
            cmdPreviousRecord.Enabled = True
            cmdNextRecord.Enabled = False
        End If
    End If
End With
End Sub
```

```
Private Sub cmdSave_Click()
Dim lngVar As Long
Dim tmpDate1 As String
Dim tmpdate2 As String
Dim tmpStr As String
Dim tmpDate As String
Dim tmpAchieve As String
```

```
On Error GoTo errSave
```

```
With rstIRA
    If cboStaffs.ListIndex = -1 Then
        ![StaffName] = ""
    Else
        ![StaffName] = cboStaffs.ItemData(cboStaffs.ListIndex)
    End If

    If Trim(txtProgressReport(0)) = "" Then
        ![PeriodFrom] = "12:00:00 PM"
    Else
        ![PeriodFrom] = Trim(txtProgressReport(0))
    End If

    If Trim(txtProgressReport(1)) = "" Then
        ![PeriodTo] = "12:00:00 PM"
    Else
        ![PeriodTo] = Trim(txtProgressReport(1))
    End If

    ![Summary] = Trim(txtProgressReport(2))
    .Update
End With

If cboGSuccessful.ListIndex = -1 Then
    tmpStr = ""
Else
    tmpStr = cboGSuccessful.ItemData(cboGSuccessful.ListIndex)
End If
```



```

If Trim(txtGCompletionDate) = "" Then
    tmpDate1 = ""
    tmpDate = ""
Else
    tmpDate1 = Trim(txtGCompletionDate)
    tmpDate = Trim(txtGCompletionDate)
End If

lngVar = rstGoal.AbsolutePosition
cnn.Execute "Update GoalAchieve Set Achievement = '" & tmpStr & "', CompletionDate = '" & tmpDate1 & _
    "' Where Episode = '" & Episode & "' And InterventionID = '" & frmIRAMenu.tmpInterv
onID & _
    "' And ConsumerID = '" & CustomerID & "' And GoalSeq = '" & GoalSeq
rstGoal.Requery
rstGoal.AbsolutePosition = lngVar
grdGoals.Rebind

If cboOSuccessful.ListIndex = -1 Then
    tmpStr = ""
    tmpAchieve = ""
Else
    tmpStr = cboOSuccessful.ItemData(cboOSuccessful.ListIndex)
    tmpAchieve = cboOSuccessful.Text
End If

If Trim(txtOCompletionDate) = "" Then
    tmpDate1 = ""
    tmpDate = ""
Else
    tmpDate1 = Trim(txtOCompletionDate)
    tmpDate = Trim(txtOCompletionDate)
End If
lngVar = rstObjectives.AbsolutePosition
cnn.Execute "Update ObjectiveAchieve Set Achievement = '" & tmpStr & "', CompletionDate = '"
pDate1 & _
    "' Where Episode = '" & Episode & "' And InterventionID = '" & frmIRAMenu.tmpInterv
onID & _
    "' And ConsumerID = '" & CustomerID & "' And ObjectiveSeq = '" & ObjectiveSeq
rstObjectives.Requery
rstObjectives.AbsolutePosition = lngVar
grdObjectives.Rebind
MsgBox "Successfully saved the information!", vbOKOnly, "Save"
Exit Sub

errSave:
MsgBox Err.Description, vbInformation + vbOKOnly, "Save"

End Sub

Private Sub Form_Load()
    Dim ProgressSeq As Long

    On Error GoTo Errhandler

    LoadControlsItems cboStaffs, False
    txtConsumerName = FirstName & " " & LastName
    txtEpisode = Episode
    txtCaseNumber = CaseNumber

    Set rstIRA = New ADODB.Recordset
    rstIRA.CursorLocation = adUseClient
    Set rstGoal = New ADODB.Recordset
    rstGoal.CursorLocation = adUseClient
    Set rstObjectives = New ADODB.Recordset
    rstObjectives.CursorLocation = adUseClient

    rstIRA.Open "Select Description From InterventionCodes Where Code = '" & frmIRAMenu.tmpInter
onID & _
    "' & "'", cnn, adOpenForwardOnly, adLockReadOnly

```

```
txtIntervention = rstIRA![Description]
rstIRA.Close
```

```
rstIRA.Open "Select ProgressSeq, StaffName, PeriodFrom, PeriodTo, Summary from ProgressReport Wh
Episode = " -
& Episode & " And InterventionID = " & frmIRAMenu.tmpInterventionID & " And ConsumerID =
```

```
CustomerID & " And CaseNumber = " & CaseNumber & " Order by ProgressSeq DESC", cnn, adOpe
namic, adLockOptimistic
```

```
With rstIRA
```

```
    If Not (.EOF And .BOF) Then
```

```
        LoadDataToControls rstIRA, cboStaffs, False
```

```
        LoadDataToControls rstIRA, txtProgressReport, True
```

```
        ProgressReportSeq ![ProgressSeq]
```

```
        txtTotalRecord = .RecordCount
```

```
        txtCurrentRecord = .AbsolutePosition
```

```
        grdGoals.RowHeight = 235
```

```
        grdObjectives.RowHeight = 235
```

```
    If .RecordCount > 1 Then
```

```
        If .AbsolutePosition = 1 Then
```

```
            cmdPreviousRecord.Enabled = False
```

```
            cmdNextRecord.Enabled = True
```

```
        ElseIf .AbsolutePosition < .RecordCount Then
```

```
            cmdPreviousRecord.Enabled = True
```

```
            cmdNextRecord.Enabled = True
```

```
        ElseIf .RecordCount = .AbsolutePosition Then
```

```
            cmdPreviousRecord.Enabled = True
```

```
            cmdNextRecord.Enabled = False
```

```
        End If
```

```
    End If
```

```
End With
```

```
Exit Sub
```

```
Errhandler:
```

```
MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    rstIRA.Close
```

```
    rstGoal.Close
```

```
    rstObjectives.Close
```

```
    Set rstIRA = Nothing
```

```
    Set rstGoal = Nothing
```

```
    Set rstObjectives = Nothing
```

```
    Set frmIRA = Nothing
```

```
End Sub
```

```
Private Sub grdGoals_Click()
```

```
    If grdGoals.SelBookmarks.Count = 0 Then Exit Sub
```

```
    rstGoal.AbsolutePosition = grdGoals.SelBookmarks(0)
```

```
With rstGoal
```

```
    If Not .EOF Then
```

```
        If .RecordCount > 1 Then
```

```
            If .AbsolutePosition = 1 Then
```

```
                cmdPreviousGoal.Enabled = False
```

```
                cmdNextGoal.Enabled = True
```

```
            ElseIf .AbsolutePosition < .RecordCount Then
```

```
                cmdPreviousGoal.Enabled = True
```

```
                cmdNextGoal.Enabled = True
```

```
            ElseIf .RecordCount = .AbsolutePosition Then
```

```
                cmdPreviousGoal.Enabled = True
```

```
                cmdNextGoal.Enabled = False
```

```
            End If
```

```
        End If
```

```

LoadDataToControls rstGoal, cboGSuccessful, False
LoadDataToControls rstGoal, txtGCompletionDate, False
txtCurrentGoal = .AbsolutePosition
txtGoal = rstGoal![Description]
GoalSeq = rstGoal![GoalSeq]
If rstObjectives.State = adStateOpen Then rstObjectives.Close
rstObjectives.Open "Select OA.GoalSeq, O.Description, OA.Achievement, OA.CompletionDate, OA.O
jectiveSeq from " & _
"ObjectiveAchieve OA, Objectives O where OA.GoalID = O.GoalID and OA.ObjectiveID = O.O
jectiveID " & _
"and OA.Episode = " & Episode & " And OA.InterventionID = " & frmIRAMenu.tmpIntervent
nID & _
" And OA.ConsumerID = " & CustomerID & " And OA.GoalSeq = " & rstGoal![GoalSeq], cnn,
dOpenDynamic, _
adLockOptimistic
With rstObjectives
If Not .EOF Then
.MoveFirst
txtObjectives = ![Description]
txtObjectiveCount = .RecordCount
txtCurrentObjective = .AbsolutePosition
ObjectiveSeq = ![ObjectiveSeq]
LoadDataToControls rstObjectives, cboOSuccessful, False
LoadDataToControls rstObjectives, txtOCompletionDate, False
Set grdObjectives.DataSource = rstObjectives

If .RecordCount > 1 Then
If .AbsolutePosition = 1 Then
cmdPreviousObjective.Enabled = False
cmdNextObjective.Enabled = True
ElseIf .AbsolutePosition < .RecordCount Then
cmdPreviousObjective.Enabled = True
cmdNextObjective.Enabled = True
ElseIf .RecordCount = .AbsolutePosition Then
cmdPreviousObjective.Enabled = True
cmdNextObjective.Enabled = False
End If
Else
cmdNextObjective.Enabled = False
cmdPreviousObjective.Enabled = False
End If
Else
cmdNextObjective.Enabled = False
cmdPreviousObjective.Enabled = False
End If
End With
Else
cmdPreviousGoal.Enabled = False
cmdNextGoal.Enabled = False
End If
End With
End Sub

```

```

Private Sub grdObjectives_Click()

If grdObjectives.SelBookmarks.Count = 0 Then Exit Sub
rstObjectives.AbsolutePosition = grdObjectives.SelBookmarks(0)

With rstObjectives
If rstObjectives.RecordCount > 1 Then
If .AbsolutePosition = 1 Then
cmdPreviousObjective.Enabled = False
cmdNextObjective.Enabled = True
ElseIf .AbsolutePosition < .RecordCount Then
cmdPreviousObjective.Enabled = True
cmdNextObjective.Enabled = True
ElseIf .RecordCount = .AbsolutePosition Then
cmdPreviousObjective.Enabled = True
cmdNextObjective.Enabled = False
End If

```

```

Else
    cmdNextObjective.Enabled = False
    cmdPreviousObjective.Enabled = False
End If

LoadDataToControls rstObjectives, cboOSuccessful, False
LoadDataToControls rstObjectives, txtOCompletionDate, False
txtObjectives = rstObjectives![Description]
txtCurrentObjective = rstObjectives.AbsolutePosition
ObjectiveSeq = ![ObjectiveSeq]
End With
End Sub

Private Sub txtGCompletionDate_Validate(Cancel As Boolean)
If Trim(txtGCompletionDate) <> "" Then
    If Not IsDate(Trim(txtGCompletionDate)) Then
        MsgBox "Please enter Date into this field!", vbInformation + vbOKOnly, "Goal Completion
e"
        txtGCompletionDate = ""
        txtGCompletionDate.SetFocus
        Cancel = True
    End If
End If
End Sub

Private Sub txtOCompletionDate_Validate(Cancel As Boolean)
If Trim(txtOCompletionDate) <> "" Then
    If Not IsDate(Trim(txtOCompletionDate)) Then
        MsgBox "Please enter Date into this field!", vbInformation + vbOKOnly, "Objective Comp
n Date"
        txtOCompletionDate = ""
        txtOCompletionDate.SetFocus
        Cancel = True
    End If
End If
End Sub

Private Sub txtProgressReport_Validate(Index As Integer, Cancel As Boolean)
If Index = 0 Or Index = 1 Then
    If txtProgressReport(Index) <> "" Then
        If Not IsDate(txtProgressReport(Index)) Then
            MsgBox "Please enter Date into this field!", vbInformation + vbOKOnly, "Report Peri
txtProgressReport(Index) = 0
            txtProgressReport(Index).SetFocus
            Cancel = True
        End If
    End If
End If
End Sub

Private Sub ProgressReportSeq(SequenceNo As Long)
If rstGoal.State = adStateOpen Then rstGoal.Close
rstGoal.Open "Select GA.GoalSeq,G.Description,GA.Achievement,GA.CompletionDate from GoalAchi
A," & _
    " Goals G Where GA.GoalID = G.GoalID and GA.Episode = " & Episode & " And GA.Intervent
= " & _
    frmIRAMenu.tmpInterventionID & " And GA.ConsumerID = " & CustomerID & " And GA.Progre
= " & _
    SequenceNo, cnn, adOpenDynamic, adLockOptimistic
If Not rstGoal.EOF Then
    If rstGoal.RecordCount > 1 Then cmdNextGoal.Enabled = True
    cmdSave.Enabled = True
    LoadDataToControls rstGoal, cboGSuccessful, False
    LoadDataToControls rstGoal, txtGCompletionDate, False
    GoalSeq = rstGoal![GoalSeq]

```

```
txtGoalCount = rstGoal.RecordCount
txtCurrentGoal = 1
txtGoal = rstGoal![Description]
Set grdGoals.DataSource = rstGoal
If rstObjectives.State = adStateOpen Then rstObjectives.Close
rstObjectives.Open "Select OA.GoalSeq,O.Description,OA.Achievement,OA.CompletionDate, OA.Ok
tiveSeq from " & _
"ObjectiveAchieve OA, Objectives O where OA.GoalID = O.GoalID and OA.ObjectiveID = O.Obj
iveID " & _
"and OA.Episode = " & Episode & " And OA.InterventionID = " & frmIRAMenu.tmpInterventio
& -
" And OA.ConsumerID = " & CustomerID & " And OA.GoalSeq = " & GoalSeq, cnn, adOpenDynar
adLockOptimistic
If Not rstObjectives.EOF Then
If rstObjectives.RecordCount > 1 Then
cmdNextObjective.Enabled = True
Else
cmdNextObjective.Enabled = False
End If

LoadDataToControls rstObjectives, choOSuccessful, False
LoadDataToControls rstObjectives, txtOCompletionDate, False
ObjectiveSeq = rstObjectives![ObjectiveSeq]
txtObjectives = rstObjectives![Description]
txtObjectiveCount = rstObjectives.RecordCount
txtCurrentObjective = 1
Set grdObjectives.DataSource = rstObjectives
End If
End If
End Sub
```

```
Public tmpInterventionID As String

Private Sub cmdClose_Click()
    Unload Me
End Sub

Private Sub cmdComputerSkills_Click()
    tmpInterventionID = "160"
    frmIRA.Show vbModal
End Sub

Private Sub cmdEmployment_Click()
    tmpInterventionID = "050"
    frmIRA.Show vbModal
End Sub

Private Sub cmdIndependentLiving_Click()
    tmpInterventionID = "013"
    frmIRA.Show vbModal
End Sub

Private Sub cmdMobilityAssessment_Click()
    tmpInterventionID = "014"
    frmIRA.Show vbModal
End Sub

Private Sub cmdSocialWork_Click()
    tmpInterventionID = "010"
    frmIRA.Show vbModal
End Sub

Private Sub Form_Load()
    Dim rst As New ADODB.Recordset

    rst.Open "Select ConsumerID From ProgressReport Where ConsumerID=" & CustomerID & " And Epis
    & _
        Episode & " And InterventionID = '014'", cnn, adOpenForwardOnly, adLockReadOnly
    cmdMobilityAssessment.Enabled = IIf(rst.EOF, False, True)
    rst.Close

    rst.Open "Select ConsumerID From ProgressReport Where ConsumerID=" & CustomerID & " And Epis
    & _
        Episode & " And InterventionID = '013'", cnn, adOpenForwardOnly, adLockReadOnly
    cmdIndependentLiving.Enabled = IIf(rst.EOF, False, True)
    rst.Close

    rst.Open "Select ConsumerID From ProgressReport Where ConsumerID=" & CustomerID & " And Epis
    & _
        Episode & " And InterventionID = '160'", cnn, adOpenForwardOnly, adLockReadOnly
    cmdComputerSkills.Enabled = IIf(rst.EOF, False, True)
    rst.Close

    rst.Open "Select ConsumerID From ProgressReport Where ConsumerID=" & CustomerID & " And Epis
    & _
        Episode & " And InterventionID = '050'", cnn, adOpenForwardOnly, adLockReadOnly
    cmdEmployment.Enabled = IIf(rst.EOF, False, True)
    rst.Close

    rst.Open "Select ConsumerID From ProgressReport Where ConsumerID=" & CustomerID & " And Epis
    & _
        Episode & " And InterventionID = '010'", cnn, adOpenForwardOnly, adLockReadOnly
    cmdSocialWork.Enabled = IIf(rst.EOF, False, True)
    rst.Close

    Set rst = Nothing
End Sub
```

```
Option Explicit
Private rstMain As ADODB.Recordset
Private rst As New ADODB.Recordset

Private Sub cboEpisode_Click()

If cboEpisode.ListIndex <> -1 Then
    Episode = Trim(cboEpisode.Text)
    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select Episode, EpisodeCloseDate From Triage Where ConsumerID = " & CustomerID & "
    And Agency = " & _
        AgencyCode & " And CaseNumber = " & CaseNumber & " And Episode = " & cboEpisode
    rst.Open
    rst.CursorLocation = adOpenForwardOnly, adLockReadOnly
    If rst![EpisodeCloseDate] = "12:00:00 PM" Then
        txtEpisodeCloseDate = ""
    Else
        txtEpisodeCloseDate = rst![EpisodeCloseDate]
    End If
    Call DisplayMainMenu
End If
End Sub

Private Sub cmdAssessments_Click()
    frmAssessments.Show vbModal
End Sub

Private Sub cmdClose_Click()
    Unload Me
End Sub

Private Sub cmdEyeData_Click()
    frmEyeData.Show vbModal
End Sub

Private Sub cmdInterventions_Click()
    frmIntervention1.Show vbModal
End Sub

Private Sub cmdIRA_Click()
    frmIRAMenu.Show vbModal
End Sub

Private Sub CmdPayer_Click()
    frmPayer.Show vbModal
End Sub

Private Sub cmdRegistration_Click()
    frmRegistration1.Show vbModal
End Sub

Private Sub cmdReport_Click()
    frmReportMenu.Show vbModal
End Sub

Private Sub cmdSearchMenu_Click()
    cboEpisode.Clear
    Episode = 0
    cmdInterventions.Enabled = False
    cmdAssessments.Enabled = False
    frmSearchMenu.Show
End Sub

Private Sub cmdTriage_Click()
    frmTriage.Show vbModal
End Sub

Private Sub Form_Activate()

txtLastName = LastName
```

```
txtFirstName = FirstName
txtCaseNumber = CaseNumber
```

```
cboEpisode.Clear
```

```
If rst.State = adStateOpen Then rst.Close
```

```
rst.Open "Select Episode, EpisodeCloseDate From Triage Where ConsumerID = " & CustomerID & " A  
Agency = " & _
```

```
AgencyCode & " And CaseNumber = " & CaseNumber & " Order by TriageSqe DESC", cn  
adOpenForwardOnly, adLockReadOnly
```

```
With rst
```

```
    If Not (.BOF And .EOF) Then
```

```
        If ([EpisodeCloseDate] = "12:00:00 PM" Or IsNull([EpisodeCloseDate])) Then  
            txtEpisodeCloseDate = ""
```

```
        Else
```

```
            txtEpisodeCloseDate = [EpisodeCloseDate]
```

```
        End If
```

```
    While Not .EOF
```

```
        cboEpisode.AddItem [Episode]
```

```
        .MoveNext
```

```
    Wend
```

```
    cboEpisode.ListIndex = 0
```

```
End If
```

```
End With
```

```
If IsNumeric(cboEpisode.Text) Then.
```

```
    Episode = CLng(Trim(cboEpisode.Text))
```

```
    Call DisplayMainMenu
```

```
End If
```

```
txtAgencyCode.SetFocus
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
On Error GoTo Errhandler
```

```
txtLastName = LastName
```

```
txtFirstName = FirstName
```

```
txtCaseNumber = CaseNumber
```

```
Set rstMain = New ADODB.Recordset
```

```
rstMain.CursorLocation = adUseClient
```

```
Set rst = New ADODB.Recordset
```

```
rst.CursorLocation = adUseClient
```

```
Exit Sub
```

```
Errhandler:
```

```
MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"
```

```
End Sub
```

```
Private Sub cmdEndOfEpisode_Click()
```

```
frmEndOfEpisode.Show vbModal
```

```
End Sub
```

```
Private Sub cmdPostEpisode_Click()
```

```
frmPostEpisode1.Show vbModal
```

```
End Sub
```

```
Private Sub DisplayMainMenu()
```

```
If rstMain.State = adStateOpen Then rstMain.Close
```

```
If rst.State = adStateOpen Then rst.Close
```

```
rstMain.Open "Select MobilityTherapy, IndependentLiving, LowVision, Insulin, DiabetesManagemen  
ychotherapy, " & _
```

```
"AcademicSkills, ComputerSkills, Employment, SocialService, ReadingService, MusicS  
1, CDC From Triage Where ConsumerID = " & _
```



```

CustomerID & " And Agency = " & AgencyCode & " And Episode = " & Episode & " And
seNumber = " & _
CaseNumber & "'"; cnn, adOpenForwardOnly, adLockReadOnly

With rstMain
  If .EOF Then
    cmdEndOfEpisode.Enabled = False
    cmdPostEpisode.Enabled = False
  Else
    If (!MobilityTherapy = "1" Or !IndependentLiving = "1" Or !Employment = "1" Or !Computer
lls = "1" Or _
    !SocialService = "1") Then
      cmdAssessments.Enabled = True
    Else
      cmdAssessments.Enabled = False
    End If

    If (!MobilityTherapy = "1" Or !IndependentLiving = "1" Or !Employment = "1" Or !Computer
lls = "1" Or _
    !SocialService = "1" Or !LowVision = "1" Or !DiabetesManagement = "1" Or !Psychoth
py = "1" Or _
    !AcademicSkills = "1" Or !Insulin = "1" Or !ReadingService = "1" Or !MusicSchool =
" Or !CDC = "1") Then
      cmdInterventions.Enabled = True
      CmdPayer.Enabled = True
    Else
      cmdInterventions.Enabled = False
      CmdPayer.Enabled = False
    End If

    rst.Open "Select ConsumerID From ProgressReport Where ConsumerID=" & CustomerID & " And
sode = " & Episode & _
    " And CaseNumber = " & CaseNumber & "'", cnn, adOpenForwardOnly, adLockReadOnly
    If rst.EOF And rst.BOF Then
      cmdIRA.Enabled = False
    Else
      cmdIRA.Enabled = True
    End If
    rst.Close

    rst.Open "Select ConsumerID from Triage Where (isdate(EpisodeCloseDate) And EpisodeClos
e <> #12:00:00 PM#) And ConsumerID=" & _
    CustomerID & " And Episode = " & Episode & " And CaseNumber = " & CaseNumber & "'",
, adOpenForwardOnly, adLockReadOnly
    If rst.EOF And rst.BOF Then
      cmdEndOfEpisode.Enabled = False
    Else
      cmdEndOfEpisode.Enabled = True
    End If
    rst.Close

    rst.Open "Select ConsumerID from QA Where ConsumerID=" & CustomerID & " And Episode = "
pisode & _
    " And CaseNumber = " & CaseNumber & "'", cnn, adOpenForwardOnly, adLockReadOnly
    If rst.EOF And rst.BOF Then
      cmdPostEpisode.Enabled = False
    Else
      cmdPostEpisode.Enabled = True
    End If
    rst.Close
  End If
End With
End Sub

Private Sub Form_Unload(Cancel As Integer)
  If Forms.Count = 1 Then
    If cnn.State = adStateOpen Then cnn.Close
    Set cnn = Nothing
  End If

```

```
If rst.State = adStateOpen Then rst.Close
Set rst = Nothing
If rstMain.State = adStateOpen Then rstMain.Close
Set rstMain = Nothing
Set frmMainMenu = Nothing
End Sub
```

```
Option Explicit
Private IsArrowKey As Integer
Private rst As ADODB.Recordset 'resordset object variable
Private rstMobility As ADODB.Recordset
Private IsVisualInfoClicked As Boolean
Public IsNewRecord As Boolean
Private IsGoalsClicked As Boolean
Public IsFileDateChecked As Boolean

Private Sub cboAgencyServices_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboAgencyServices(Index).ListIndex = -1
    End If
End Sub

Private Sub cboAgree_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboAgree(Index).ListIndex = -1
    End If
End Sub

Private Sub cboAgree_LostFocus(Index As Integer)
    If CheckForTab Then
        If Index = 38 Then
            tabMobility.Tab = 2
        ElseIf Index = 28 Then
            tabMobility.Tab = 3
        ElseIf Index = 20 Then
            tabMobility.Tab = 4
        ElseIf Index = 15 Then
            tabMobility.Tab = 5
        ElseIf Index = 6 Then
            tabMobility.Tab = 6
        ElseIf Index = 0 Then
            tabMobility.Tab = 7
        End If
    End If
End Sub

Private Sub cboAgree_Validate(Index As Integer, Cancel As Boolean)
    If Index = 0 Or Index = 1 Or Index = 2 Or Index = 3 Or Index = 4 Or Index = 5 Then
        If cboYesNo(18).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 6 Or Index = 7 Or Index = 8 Then
        If cboYesNo(19).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 9 Or Index = 10 Or Index = 11 Or Index = 12 Or Index = 13 Or Index = 14 Or Index = 15 Or Index = 16 Then
        If cboYesNo(20).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 17 Then
        If cboYesNo(21).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 18 Or Index = 19 Then
        If cboYesNo(22).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 20 Or Index = 21 Or Index = 22 Or Index = 23 Or Index = 24 Or Index = 25 Then
        If cboYesNo(23).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 26 Then
        If cboYesNo(24).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 27 Or Index = 28 Or Index = 29 Or Index = 30 Or Index = 31 Or Index = 32 Or Index = 33 Or Index = 34 Then
        If cboYesNo(25).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 35 Then
        If cboYesNo(26).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 42 Or Index = 43 Then
        If cboYesNo(27).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 36 Or Index = 37 Or Index = 38 Or Index = 39 Or Index = 40 Or Index = 41 Then
        If cboYesNo(28).ListIndex <> 0 Then cboAgree(Index).ListIndex = -1
    End If
End Sub
```

```
Private Sub cboConsumerRating_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboConsumerRating(Index).ListIndex = -1
End If
End Sub

Private Sub cboDifficulty_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboDifficulty(Index).ListIndex = -1
End If
End Sub

Private Sub cboEquipment_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboEquipment(Index).ListIndex = -1
End If
End Sub

Private Sub cboExternalServices_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboExternalServices(Index).ListIndex = -1
End If
End Sub

Private Sub cboFactors_Click(Index As Integer)

If IsArrowKey > 0 Then Exit Sub

If Index = 4 Then
    CountScore
    CountGoals
End If
End Sub

Private Sub cboFactors_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub

Private Sub cboFactors_LostFocus(Index As Integer)

If CheckForTab Then
    If Index = 4 Then
        CountScore
        CountGoals
    End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboObserveResponse_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboObserveResponse(Index).ListIndex = -1
End If
```

End Sub

Private Sub cboRating\_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then  
    cboRating(Index).ListIndex = -1  
End If

End Sub

Private Sub cboServiceLevel\_Click(Index As Integer)

If IsArrowKey > 0 Then Exit Sub

If Index = 0 Then  
    If cboServiceLevel(Index).ListIndex = -1 Or cboServiceLevel(Index).ListIndex = 0 Then  
        txtAssessment(14) = ""  
    Else  
        cboTimeFrame.ListIndex = cboServiceLevel(Index).ListIndex - 1  
        txtAssessment(14) = cboTimeFrame.Text  
        Call CountGoals  
    End If

    tabMobility.Tab = 9

End If

End Sub

Private Sub cboServiceLevel\_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then  
    IsArrowKey = KeyCode  
ElseIf KeyCode = vbKeyDown Then  
    IsArrowKey = KeyCode  
ElseIf KeyCode = vbKeyRight Then  
    IsArrowKey = KeyCode  
ElseIf KeyCode = vbKeyLeft Then  
    IsArrowKey = KeyCode  
End If

End Sub

Private Sub cboServiceLevel\_LostFocus(Index As Integer)

If Index = 0 Then  
    If CheckForTab Then  
        If cboServiceLevel(Index).ListIndex = -1 Or cboServiceLevel(Index).ListIndex = 0 Then  
            txtAssessment(14) = ""  
        Else  
            cboTimeFrame.ListIndex = cboServiceLevel(Index).ListIndex - 1  
            txtAssessment(14) = cboTimeFrame.Text  
            Call CountGoals  
        End If

    tabMobility.Tab = 9

End If

End Sub

If IsArrowKey > 0 Then IsArrowKey = 0  
End Sub

Private Sub cboStaffs\_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then  
    cboStaffs.ListIndex = -1  
End If

End Sub

Private Sub cboYesNo\_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then  
    cboYesNo(Index).ListIndex = -1

```
End If
End Sub
```

```
Private Sub cboYesNo_LostFocus(Index As Integer)
If Index = 17 Or Index = 16 Or Index = 15 Or Index = 14 Or Index = 13 Or Index = 12 Or Index :
Or _
Index = 10 Or Index = 9 Or Index = 8 Then
If CheckForTab Then
If cboYesNo(Index).ListIndex = 0 Then
cboObserveResponse(Index - 8).SetFocus
Else
If Index = 8 Then
txtAssessment(5).SetFocus
Else
cboYesNo(Index - 1).SetFocus
End If
End If
End If
End If
End Sub
```

```
Private Sub cmdClear_Click()

IsArrowKey = 20          'Avoid Conbo Click event
IsNewRecord = True
clearData txtAssessment, True
clearData lstDestinations, False
clearData chkGoals, True
clearData cboYesNo, True
clearData cboAgree, True
clearData cboExternalServices, True
clearData cboAgencyServices, True
clearData cboEquipment, True
clearData cboServiceLevel, True
clearData cboFactors, True
clearData cboStrategiesLearning, False
clearData cboObserveResponse, True
clearData cboConsumerRating, True
clearData cboRating, True
clearData cboDifficulty, True
clearData cboStaffs, False

If Not rstMobility.BOF Then rstMobility.MoveFirst
txtCurrentRecord = ""
cmdPrint.Enabled = False
cmdSave.Enabled = True
tabMobility.Tab = 0
IsArrowKey = 0
End Sub
```

```
Private Sub cmdClose_Click()
Unload Me
End Sub
```

```
Private Sub cmdHistorical_Click()
IsVisualInfoClicked = True
frmEyeData.Show vbModal
End Sub
```

```
Private Sub cmdNext_Click()
If tabMobility.Tab = 9 Then
MsgBox "It is last page for Services Agreement already!", vbOKOnly, "Next Page"
Else
tabMobility.Tab = tabMobility.Tab + 1
frmMobility.Caption = getTabCaption(tabMobility.Tab)
End If
End Sub
```

```
Private Sub cmdNextRecord_Click()
```

```
With rstMobility
    .MoveNext
    If .AbsolutePosition = 1 Then
        cmdPreviousRecord.Enabled = False
        cmdNextRecord.Enabled = True
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = False
    End If

    txtCurrentRecord = .AbsolutePosition
    txtSequencyNo = ![AssessmentSequence]
    LoadDataToControls rstMobility, lstDestinations, False
    LoadDataToControls rstMobility, cboStrategiesLearning, False
    LoadDataToControls rstMobility, cboStaffs, False
    LoadDataToControls rstMobility, txtAssessment, True
    LoadDataToControls rstMobility, chkGoals, True
    LoadDataToControls rstMobility, cboYesNo, True
    LoadDataToControls rstMobility, cboAgree, True
    LoadDataToControls rstMobility, cboExternalServices, True
    LoadDataToControls rstMobility, cboAgencyServices, True
    LoadDataToControls rstMobility, cboEquipment, True
    LoadDataToControls rstMobility, cboServiceLevel, True
    LoadDataToControls rstMobility, cboObserverResponse, True
    LoadDataToControls rstMobility, cboConsumerRating, True
    LoadDataToControls rstMobility, cboRating, True
    LoadDataToControls rstMobility, cboDifficulty, True
    LoadDataToControls rstMobility, cboFactors, True

    If ![FinishDate] = "12:00:00 PM" Then
        cmdSave.Enabled = True
        cmdPrint.Enabled = False
        cmdClear.Enabled = False
    Else
        cmdSave.Enabled = False
        cmdPrint.Enabled = True
        cmdClear.Enabled = True
    End If
End With

tabMobility.Tab = 0
txtCurrentRecord.SetFocus
End Sub

Private Sub cmdPrevious_Click()
    If tabMobility.Tab = 0 Then
        MsgBox "It is first page for Self Assessment already!", vbOKOnly, "Previous Page"
    Else
        tabMobility.Tab = tabMobility.Tab - 1
        frmMobility.Caption = getTabCaption(tabMobility.Tab)
    End If
End Sub

Private Sub cmdPreviousRecord_Click()
    With rstMobility
        .MovePrevious
        If .AbsolutePosition = 1 Then
            cmdPreviousRecord.Enabled = False
            cmdNextRecord.Enabled = True
        ElseIf .AbsolutePosition < .RecordCount Then
            cmdPreviousRecord.Enabled = True
            cmdNextRecord.Enabled = True
        ElseIf .RecordCount = .AbsolutePosition Then
            cmdPreviousRecord.Enabled = True
            cmdNextRecord.Enabled = False
        End If
    End With
End Sub
```

```

If ![FinishDate] = "12:00:00 PM" Then
    cmdSave.Enabled = True
    cmdPrint.Enabled = False
    cmdClear.Enabled = False
Else
    cmdSave.Enabled = False
    cmdPrint.Enabled = True
    cmdClear.Enabled = True
End If

```

```

txtCurrentRecord = .AbsolutePosition
txtSequencyNo = ![AssessmentSequence]
LoadDataToControls rstMobility, lstDestinations, False
LoadDataToControls rstMobility, cboStrategiesLearning, False
LoadDataToControls rstMobility, cboStaffs, False
LoadDataToControls rstMobility, txtAssessment, True
LoadDataToControls rstMobility, chkGoals, True
LoadDataToControls rstMobility, cboYesNo, True
LoadDataToControls rstMobility, cboAgree, True
LoadDataToControls rstMobility, cboExternalServices, True
LoadDataToControls rstMobility, cboAgencyServices, True
LoadDataToControls rstMobility, cboEquipment, True
LoadDataToControls rstMobility, cboServiceLevel, True
LoadDataToControls rstMobility, cboObserveResponse, True
LoadDataToControls rstMobility, cboConsumerRating, True
LoadDataToControls rstMobility, cboDifficulty, True
LoadDataToControls rstMobility, cboRating, True
LoadDataToControls rstMobility, cboFactors, True

```

```
End With
```

```

tabMobility.Tab = 0
txtCurrentRecord.SetFocus
End Sub

```

```

Private Sub cmdPrint_Click()
Dim i As Integer

```

```

If Printers.Count = 0 Then
    MsgBox "No printer found associating with this computer. Please add a printer to it and then you can print out the Service agreement! ", _
        vbOKOnly, "Printer"
Else
    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select * from MobilityAgreement Where AssessmentSequence = " & txtSequencyNo, cnn
    dOpenKeyset, adLockOptimistic
    If Not rst.EOF Then ' if there an existing report, delete it then create a new report
        rst.Delete
    End If
    rst.Close
    If rst.State = adStateOpen Then rst.Close
    rst.Open "Select * from MobilityAgreement Where AssessmentSequence = " & txtSequencyNo, cnn
    dOpenKeyset, adLockOptimistic
    With rst
        If .BOF And .EOF Then
            .AddNew
            ![CaseNumber] = txtCaseNumber
            ![Episode] = txtEpisode
            ![AssessmentSequence] = txtSequencyNo
            ![ConsumerName] = txtConsumerName
            ![AssessmentDate] = txtAssessment(0)
            ![AssessorName] = cboStaffs.Text
            For i = 0 To chkGoals.Count - 1
                If chkGoals(i).Value = 0 Then
                    .Fields(chkGoals(i).Tag) = ""
                Else
                    .Fields(chkGoals(i).Tag) = "Yes"
                End If
            Next
        End If
    End With

```



```

For i = 0 To 7
    .Fields(cboYesNo(i).Tag) = cboYesNo(i).Text
Next

For i = 0 To cboEquipment.Count - 1
    .Fields(cboEquipment(i).Tag) = cboEquipment(i).Text
Next

For i = 0 To cboAgencyServices.Count - 1
    .Fields(cboAgencyServices(i).Tag) = cboAgencyServices(i).Text
Next

For i = 0 To cboExternalServices.Count - 1
    .Fields(cboExternalServices(i).Tag) = cboExternalServices(i).Text
Next

.Fields("TimeFrame") = cboTimeFrame.Text
.Fields("Comments") = txtAssessment(8).Text
.Update
End If
End With
Set MobilityAgreement.DataSource = rst
MobilityAgreement.PrintReport
End If
End Sub

Private Sub cmdSave_Click()
Dim i As Integer
Dim colMobility As New Collection

On Error GoTo errSave

colMobility.Add cboRating
colMobility.Add cboAgree
colMobility.Add lstDestinations
colMobility.Add cboStrategiesLearning
colMobility.Add cboStaffs
colMobility.Add txtAssessment
colMobility.Add chkGoals
colMobility.Add cboYesNo
colMobility.Add cboExternalServices
colMobility.Add cboAgencyServices
colMobility.Add cboEquipment
colMobility.Add cboServiceLevel
colMobility.Add cboFactors
colMobility.Add cboObserveResponse
colMobility.Add cboConsumerRating
colMobility.Add cboDifficulty

If IsDate(txtAssessment(0)) Then
    ' if filing assessment make sure learning str
    ies
    If cboStrategiesLearning.ListIndex = -1 Then ' and factors are there
        tabMobility.Tab = 8
        cboStrategiesLearning.SetFocus
        MsgBox "You have to select one of items from Strategies for Learning!", vbOKOnly, "Stra
Learning"
        Exit Sub
    End If

    For i = 0 To cboFactors.Count - 1
        If cboFactors(i).ListIndex = -1 Then
            tabMobility.Tab = 8
            cboFactors(i).SetFocus
            MsgBox "You have to select one of items from Factors!", vbOKOnly, "Factors"
            Exit Sub
        End If
    Next
End If

If SaveDataToDB(frmMobility, rstMobility, colMobility, "014", False) Then
    'Call pub

```

unction to save data to DB

MsgBox "Successfully saved the Assessment information!", vbOKOnly, "Save"

If IsDate(txtAssessment(0)) Then

cnn.Execute "Update Interventions Set PreConsumerScore = '" & txtConsumerScore & "', Pre  
viderScore = '" & \_  
txtProviderScore & "' Where ConsumerID = '" & CustomerID & "' And Agency = '" & AgencyC  
& -  
" And Episode = '" & Episode & "' And CaseNumber = '" & CaseNumber & -  
" And InterventionID = '014' And (EndDate = #12:00:00 PM# or EndDate IS NULL)"  
cnn.Execute "UPDATE MobilityAssessments AS a INNER JOIN Interventions AS b ON (a.CaseNum  
= b.CaseNumber)" & -  
" AND (a.Episode = b.Episode) AND (a.ConsumerID = b.ConsumerID) SET a.InterventionSeq  
ce = b.InterventionSequence" & -  
" WHERE b.InterventionID = '014' and b.CaseNumber = '" & CaseNumber & "' AND (b.EndDa  
= #12:00:00PM# OR b.EndDate IS NULL)" & -  
" AND a.InterventionSequence = 0"

End If

frmMobility.Refresh

If IsNewRecord Then

rstMobility.Requery

txtSequencyNo = rstMobility![AssessmentSequence]

End If

frmMobility.Refresh

IsNewRecord = False

Set colMobility = Nothing

End If

Exit Sub

errSave:

MsgBox Err.Description & Err.Number, vbCritical + vbOKOnly, "Save"

End Sub

Private Sub Form\_Activate()

On Error GoTo Errhandler

If IsVisualInfoClicked Then

IsVisualInfoClicked = False

Exit Sub

End If

txtConsumerName = FirstName & " " & LastName

txtPhone = PhoneNo

txtAddress = Address

txtCaseNumber = CaseNumber

txtEpisode = Episode

txtIntervention = "Mobility Assessment"

Set rst = New ADODB.Recordset

'create recordset object

rst.CursorLocation = adUseClient

LoadControlsItems cboStaffs, False

'Add every description from database to combobox of  
tbox

Set rstMobility = New ADODB.Recordset

rstMobility.CursorLocation = adUseClient

rstMobility.Open "Select \* From MobilityAssessments Where ConsumerID = '" & CustomerID & -  
" And Agency = '" & AgencyCode & "' And Episode = '" & Episode & "' And CaseNumber = '" & -  
CaseNumber & "' And InterventionID = '014' Order By AssessmentSequence DESC", cnn, -  
adOpenKeyset, adLockOptimistic

With rstMobility

txtRecordNo = .RecordCount

If Not (.EOF And .BOF) Then

'Have record for this client and load it t

m

txtCurrentRecord = .AbsolutePosition

txtSequencyNo = ![AssessmentSequence]

LoadDataToControls rstMobility, lstDestinations, False

```

LoadDataToControls rstMobility, cboStrategiesLearning, False
LoadDataToControls rstMobility, cboStaffs, False
LoadDataToControls rstMobility, txtAssessment, True
LoadDataToControls rstMobility, chkGoals, True
LoadDataToControls rstMobility, cboYesNo, True
LoadDataToControls rstMobility, cboAgree, True
LoadDataToControls rstMobility, cboExternalServices, True
LoadDataToControls rstMobility, cboAgencyServices, True
LoadDataToControls rstMobility, cboEquipment, True
LoadDataToControls rstMobility, cboServiceLevel, True
LoadDataToControls rstMobility, cboObserverResponse, True
LoadDataToControls rstMobility, cboConsumerRating, True
LoadDataToControls rstMobility, cboRating, True
LoadDataToControls rstMobility, cboDifficulty, True
LoadDataToControls rstMobility, cboFactors, True

If .RecordCount > .AbsolutePosition Then cmdNext.Enabled = True

If ![FinishDate] <> "12:00:00 PM" And IsDate(![FinishDate]) Then
    cmdSave.Enabled = False
    cmdPrint.Enabled = True
    cmdClear.Enabled = True
End If

If .RecordCount <= .1 Then
    cmdNextRecord.Enabled = False
    cmdPreviousRecord.Enabled = False
ElseIf .RecordCount >= 2 Then
    cmdPreviousRecord.Enabled = False
    cmdNextRecord.Enabled = True
End If
Else
    cmdNextRecord.Enabled = False
    cmdPreviousRecord.Enabled = False
    txtCurrentRecord = 0
End If
End With

tabMobility.Tab = 0
txtConsumerName.SetFocus
Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"
End Sub

Private Sub Form_Unload(Cancel As Integer)
Dim response As Integer

If IsFileDateChecked Then
    If rst.State = adStateOpen Then rst.Close
    Set rst = Nothing
    tabMobility.Tab = 0
    Set frmMobility = Nothing
Else
    If Trim(txtAssessment(0)) = "" And cmdSave.Enabled Then
        response = MsgBox("Have you finished O&M Assessment ?", vbQuestion + vbYesNo, "File Date")
        If response = vbYes Then
            MsgBox "Please enter the File Date !", vbOKOnly, "File Date"
            txtAssessment(0).SetFocus
            Cancel = True
        Else
            If rst.State = adStateOpen Then rst.Close
            Set rst = Nothing
            tabMobility.Tab = 0
            Set frmMobility = Nothing
        End If
    ElseIf IsDate(Trim(txtAssessment(0))) And cmdSave.Enabled Then
        response = MsgBox("Are you sure you finished the O&M Assessment?", vbQuestion + vbYesNo, "File Date")
    End If

```

```

    If response = vbYes Then
        response = MsgBox("Please use save button to to save the assessment information", vbOK
ly, "File Date")
        cmdSave.SetFocus
    ElseIf cmdSave.Enabled = True Then
        txtAssessment(0) = ""
    End If
    ElseIf IsDate(Trim(txtAssessment(0))) And Not cmdSave.Enabled Then
        If rst.State = adStateOpen Then rst.Close
        Set rst = Nothing
        tabMobility.Tab = 0
        Set frmMobility = Nothing
    End If
End If
End Sub

Private Sub cmdFactors_Click()
Dim response As Integer

tabMobility.Tab = 7
response = MsgBox("Rate on a scale of -3 to +3 where" & Chr(13) & Chr(10) & _
    "-3 will require a significant decrease in training time" & Chr(13) & Chr(10) & _
    "-2 will require a moderate decrease in training time" & Chr(13) & Chr(10) & _
    "-1 will require a minimal decrease in training time" & Chr(13) & Chr(10) & _
    "0 will have no impact on training time" & Chr(13) & Chr(10) & _
    "+1 will require a minimal increase in training time" & Chr(13) & Chr(10) & _
    "+2 will require a moderate increase in training time" & Chr(13) & Chr(10) & _
    "+3 will require a significant increase in training time", vbOKOnly, _
    "Additional Factors")
If response = vbOK Then
    tabMobility.Tab = 8
    cboFactors(0).SetFocus
End If
End Sub

Private Sub cmdGlassDevices_Click()
MsgBox "What knid of glasses or low vision devices do you use? (Self-reported. May choose more
an one)", _
    vbOKOnly, "Glass or LV Devices"
End Sub

Private Sub cmdObserverInform_Click()
MsgBox "If problem affects safety, indicate problem, inform consumer and record their response
    vbOKOnly, "Observer. and Inform Comsumer"
End Sub

Private Sub cmdGoal_Click()
Dim response As Integer

IsGoalsClicked = True
tabMobility.Tab = 8
response = MsgBox("Based on the assessment, the following goals are recommended.", _
    vbOKOnly, "Goals")
If response = vbOK Then
    tabMobility.Tab = 9
    chkGoals(0).SetFocus
    CountGoals
End If
IsGoalsClicked = False
End Sub

Private Sub cmdObserveInform_Click()
Dim response As Integer

MsgBox "If problem affects safety, indicate problem, inform consumer and record their respons
vbOKOnly, "Observe and Inform Consumer"
tabMobility.Tab = 6
cmdObserveTrain(9).SetFocus
tabMobility.Tab = 7

```

```
cboYesNo(17).SetFocus
End Sub
```

```
Private Sub cmdObserveInstruction_Click()
```

```
MsgBox " Both the provider and consumer rate the following items." & Chr(10) & Chr(13) & Chr(13)
& - "When asking the consumer, use the standard 4 point rating scale for each questions:" & -
Chr(13) & Chr(10) & " Great deal of difficulty 1" & Chr(13) & Chr(10) & -
" Some difficulty 2" & Chr(13) & Chr(10) & " Little difficulty 3" & Chr(13) &
Chr(10) & - " No difficulty 4" & Chr(13) & Chr(10) & " Refuses 4" & Chr(13) & Chr(13) & (
r(10) & -
"Provider use the standard 10 points scale: 1 = major problem, 10 = no problem, N/A", -
vbOKOnly, "Observe and Train"
frmMobility.Refresh
tabMobility.Tab = 2
tabMobility.Tab = 1
cmdObserveTrain(0).SetFocus
End Sub
```

```
Private Sub cmdObserveTrain_Click(Index As Integer)
```

```
If Index = 0 Then
MsgBox "4. How much difficulty do you have walking with another person leading you?" & Chr(1
& Chr(10) & -
"(If no one guiding them, mark their response with a 1)", -
vbOKOnly, "Self assessment question"
tabMobility.Tab = 2
tabMobility.Tab = 1
cboConsumerRating(9).SetFocus
ElseIf Index = 1 Then
MsgBox "5. How much difficulty do you have getting around your own home?", vbOKOnly, -
"Self assessment question"
tabMobility.Tab = 2
tabMobility.Tab = 1
cboConsumerRating(8).SetFocus
ElseIf Index = 2 Then
MsgBox "6. How much difficulty do you have asking questions if you're lost or can't find som
hing?", -
vbOKOnly, "Self assessment question"
tabMobility.Tab = 3
tabMobility.Tab = 2
cboConsumerRating(7).SetFocus
ElseIf Index = 3 Then
MsgBox "7. How much difficulty do you have going down steps?", vbOKOnly, "Self assessment q
tion"
tabMobility.Tab = 4
tabMobility.Tab = 3
cboConsumerRating(6).SetFocus
ElseIf Index = 4 Then
MsgBox "8. How much difficulty do you have walking on a sidewalk?", vbOKOnly, "Self assessm
question"
tabMobility.Tab = 4
tabMobility.Tab = 3
cboConsumerRating(5).SetFocus
ElseIf Index = 5 Then
MsgBox "9. (For cane user only) How much difficulty do you have finding curbs or cracks in
sidewalk with your cane?", vbOKOnly, "Self assessment question"
tabMobility.Tab = 3
tabMobility.Tab = 4
cboConsumerRating(4).SetFocus
ElseIf Index = 6 Then
MsgBox "10. (For dog guide users only) How much difficulty do you have travelling with and
trolling your dog guide?", vbOKOnly, "Self assessment question"
tabMobility.Tab = 3
tabMobility.Tab = 4
cboConsumerRating(3).SetFocus
ElseIf Index = 7 Then
MsgBox "11. How much difficulty do you have crossing safely?", vbOKOnly, "Self assessment q
```

```

tion"
    tabMobility.Tab = 3
    tabMobility.Tab = 4
    cboConsumerRating(2).SetFocus
ElseIf Index = 8 Then
    MsgBox "12. When you're in an unfamiliar building, for example," & _
        " a store or a doctor's office, how much difficulty to you have getting around safely?"
    vbOKOnly, "Self assessment question"
    tabMobility.Tab = 6
    tabMobility.Tab = 5
    cboConsumerRating(1).SetFocus
ElseIf Index = 9 Then
    MsgBox "13. When using public transportation or para-transit, how much difficulty do you hav
using this transportation by yourself", _
        vbOKOnly, "Self assessment question"
    tabMobility.Tab = 5
    tabMobility.Tab = 6
    cboConsumerRating(0).SetFocus
End If
End Sub

Private Sub lstDestinations_KeyUp(KeyCode As Integer, Shift As Integer)
Dim i As Integer

If KeyCode = vbKeyBack Then
    If lstDestinations.SelCount = 1 Then
        lstDestinations.Selected(lstDestinations.ListIndex) = False
    ElseIf lstDestinations.ListCount > 1 Then
        For i = lstDestinations.ListCount - 1 To 0 Step -1
            If lstDestinations.Selected(i) Then
                lstDestinations.Selected(i) = False
            End If
        Next
    End If
End If
End Sub

Private Sub txtAssessment_LostFocus(Index As Integer)
Dim response As Integer

If CheckForTab Then
    If Index = 6 Then
        tabMobility.Tab = 1
    ElseIf Index = 5 Then
        tabMobility.Tab = 8
    End If
End If

If Index = 0 And CheckForTab Then
    If Trim(txtAssessment(0)) = "" And cmdSave.Enabled Then
        response = MsgBox("Have you finished O&M Assessment ?", vbQuestion + vbYesNo, "File Date
e Date")
        If response = vbYes Then
            MsgBox "Please enter the File Date !", vbOKOnly, "File Date"
            txtAssessment(0).SetFocus
        End If
    ElseIf IsDate(Trim(txtAssessment(0))) And cmdSave.Enabled Then
        response = MsgBox("Are you sure you finish the O&M Assessment?", vbQuestion + vbYesNo, "
e Date")
        If response = vbYes Then
            MsgBox "Please use Save button to to save the Assessment information ?", vbOKOnly, "F
Date"
            cmdSave.SetFocus
        Else
            txtAssessment(0) = ""
        End If
    End If
    IsFileDateChecked = True
End If
End Sub

```

```
Private Function getTabCaption(tabIndex As Integer) As String
```

```
Select Case tabIndex
```

```
Case 0
```

```
    getTabCaption = "A. Self Assessment"
```

```
Case 1
```

```
    getTabCaption = "B. Observe And Train"
```

```
Case 2
```

```
    getTabCaption = "C. Observe And Train"
```

```
Case 3
```

```
    getTabCaption = "D. Observe And Train"
```

```
Case 4
```

```
    getTabCaption = "E. Observe And Train"
```

```
Case 5
```

```
    getTabCaption = "F. Observe And Train"
```

```
Case 6
```

```
    getTabCaption = "G. Observe And Train"
```

```
Case 7
```

```
    getTabCaption = "H. Observe And Inform"
```

```
Case 8
```

```
    getTabCaption = "I. Learning Strategy and Factors"
```

```
Case 9
```

```
    getTabCaption = "J. O&M Services Agreement"
```

```
End Select
```

```
End Function
```

```
Private Sub CountScore()
```

```
Dim i As Integer
```

```
Dim tmpValue As Integer
```

```
Dim taskA As Long
```

```
Dim taskB As Long
```

```
Dim taskD As Long
```

```
Dim taskE As Long
```

```
Dim avgScore As Single
```

```
If cboStrategiesLearning.ListIndex = -1 Then
```

```
    tabMobility.Tab = 8
```

```
    cboStrategiesLearning.SetFocus
```

```
    MsgBox "You had to select one of items from Strategy Learning!", vbOKOnly, "Strategy Learn
```

```
Exit Sub
```

```
End If
```

```
For i = 0 To cboFactors.Count - 1
```

```
    If cboFactors(i).ListIndex = -1 Then
```

```
        tabMobility.Tab = 8
```

```
        cboFactors(i).SetFocus
```

```
        MsgBox "You had to select one of items from Factors!", vbOKOnly, "Factors"
```

```
Exit Sub
```

```
End If
```

```
Next
```

```
For i = 0 To cboDifficulty.Count - 1 'count the score for task A
```

```
    If cboDifficulty(i).ListIndex = -1 Then
```

```
        taskA = taskA + 4
```

```
    Else
```

```
        tmpValue = cboDifficulty(i).ItemData(cboDifficulty(i).ListIndex)
```

```
        If tmpValue = 5 Then
```

```
            'User select "Refused", count 1 for this item
```

```
            taskA = taskA + 1
```

```
        Else
```

```
            taskA = taskA + tmpValue
```

```
        End If
```

```
    End If
```

```
Next
```

```
For i = 0 To cboConsumerRating.Count - 1 'count the score for task A
```

```
    If cboConsumerRating(i).ListIndex = -1 Then
```

```
        taskA = taskA + 4
```

```
    Else
```

```
        tmpValue = cboConsumerRating(i).ItemData(cboConsumerRating(i).ListIndex)
```

```

        If tmpValue = 5 Then
            taskA = taskA + 0
        Else
            taskA = taskA + tmpValue
        End If
    End If
Next

tmpValue = 0
For i = 0 To cboRating.Count - 1
    'count the score for Task B
    If cboRating(i).ListIndex = -1 Then 'user does'nt select one, count 10 for this item
        taskB = taskB + 10
    Else
        tmpValue = cboRating(i).ItemData(cboRating(i).ListIndex)
        If tmpValue = 11 Then 'User select "Refused", count 1 for this item
            taskB = taskB + 1
        Else
            taskB = taskB + tmpValue
        End If
    End If
Next

taskD = cboStrategiesLearning.ItemData(cboStrategiesLearning.ListIndex)

tmpValue = 0
For i = 0 To cboFactors.Count - 1
    Select Case cboFactors(i).ItemData(cboFactors(i).ListIndex)
        Case -3: tmpValue = 120
        Case -2: tmpValue = 110
        Case -1: tmpValue = 105
        Case 0: tmpValue = 91
        Case 1: tmpValue = 80
        Case 2: tmpValue = 60
        Case 3: tmpValue = 25
    End Select
    taskE = taskE + tmpValue
Next

txtProviderScore = Trim(taskB * 0.23)
txtConsumerScore = Trim(taskA * 1.78)
avgScore = ((taskA * 1.78) + (taskB * 0.23) + taskD + (taskE / 5)) / 4
txtAssessment(3) = CLng(avgScore)
txtAssessment(9) = Format(taskA * 1.78, "###")
txtAssessment(10) = Format(((taskB * 100) / 440), "###")
txtAssessment(11) = Format(taskD, "###")
txtAssessment(12) = Format(taskE / 5, "###")
txtAssessment(13) = Format(avgScore, "###")

Select Case CLng(avgScore)
    Case 0 To 60: txtAssessment(1) = "Level 3"
    Case 61 To 80: txtAssessment(1) = "Level 2"
    Case 81 To 90: txtAssessment(1) = "Level 1"
    Case 91 To 100: txtAssessment(1) = "No Service Needed"
End Select
End Sub

Private Sub tabMobility_Click(PreviousTab As Integer)
    Dim i As Integer

    If tabMobility.Tab = 0 Then
        cboStaffs.SetFocus
        frmMobility.Caption = "A. Self Assessment"
    ElseIf tabMobility.Tab = 1 Then
        cmdObserveInstruction.SetFocus
        frmMobility.Caption = "B. Observe And Train"
    ElseIf tabMobility.Tab = 2 Then
        cboRating(37).SetFocus
        frmMobility.Caption = "C. Observe And Train"
    ElseIf tabMobility.Tab = 3 Then
        cboRating(27).SetFocus
    End If
End Sub

```



```

frmMobility.Caption = "D. Observer And Train"
ElseIf tabMobility.Tab = 4 Then
    cmdObserveTrain(5).SetFocus
    frmMobility.Caption = "E. Observer And Train"
ElseIf tabMobility.Tab = 5 Then
    cboRating(14).SetFocus
    frmMobility.Caption = "F. Observer And Train"
ElseIf tabMobility.Tab = 6 Then
    cmdObserveTrain(0).SetFocus
    frmMobility.Caption = "G. Observer And Train"
ElseIf tabMobility.Tab = 7 Then
    cmdObserveInform.SetFocus
    frmMobility.Caption = "H. Observer And Inform Consumer"
ElseIf tabMobility.Tab = 8 Then
    cboStrategiesLearning.SetFocus
    frmMobility.Caption = "I. Learning Strategies, Factors and Recommended Service Level"
ElseIf tabMobility.Tab = 9 Then
    If Not IsGoalsClicked Then
        cmdGoal.SetFocus
        frmMobility.Caption = "J. Services Agreement"

        If cboServiceLevel(0).ListIndex = -1 Or cboServiceLevel(0).ListIndex = 0 Then
            txtAssessment(14) = ""
        Else
            cboTimeFrame.ListIndex = cboServiceLevel(0).ListIndex - 1
            txtAssessment(14) = cboTimeFrame.Text
            Call CountGoals
        End If
    End If
End If
End Sub

Private Sub txtAssessment_Validate(Index As Integer, Cancel As Boolean)
    If Index = 2 Or Index = 7 Or Index = 0 Then
        If Trim(txtAssessment(Index)) <> "" Then
            If Not IsDate(Trim(txtAssessment(Index))) Then
                MsgBox "Please enter date into this field !", vbOKOnly, "Date"
                txtAssessment(Index) = ""
                Cancel = True
            End If
        End If
    End If
End Sub

Private Sub txtCurrentRecord_LostFocus()
    If CheckForTab Then
        cboStaffs.SetFocus
        frmMobility.Caption = "Self Assessment"
    End If
End Sub

Private Sub CountGoals()
    Dim i As Integer
    For i = 0 To cboAgree.Count - 1
        Select Case i
            Case 0 To 5
                If cboAgree(i).Text = "Yes" Then chkGoals(10) = 1
            Case 6 To 8
                If cboAgree(i).Text = "Yes" Then chkGoals(9) = 1
            Case 9 To 16
                If cboAgree(i).Text = "Yes" Then chkGoals(8) = 1
            Case 17
                If cboAgree(i).Text = "Yes" Then chkGoals(7) = 1
            Case 18 To 19
                If cboAgree(i).Text = "Yes" Then chkGoals(6) = 1
            Case 20 To 25
                If cboAgree(i).Text = "Yes" Then chkGoals(5) = 1
        End Select
    Next i
End Sub

```

```

Case 26
  If cboAgree(i).Text = "Yes" Then chkGoals(4) = 1
Case 27 To 34
  If cboAgree(i).Text = "Yes" Then chkGoals(3) = 1
Case 35
  If cboAgree(i).Text = "Yes" Then chkGoals(2) = 1
Case 36 To 41
  If cboAgree(i).Text = "Yes" Then chkGoals(1) = 1
Case 42 To 43
  If cboAgree(i).Text = "Yes" Then chkGoals(0) = 1
End Select
Next
End Sub

```

```
Option Explicit
Private rsScores As ADODB.Recordset
```

```
Private Sub Form_Load()
```

```
Dim PreSelf As Single
Dim PreProv As Single
Dim PreServiceScore As Single
Dim PostSelf As Single
Dim PostProv As Single
Dim PostServiceScore As Single
Dim PercentChange As Single
Dim AssessmentText(5) As String
Dim InterventionID(5) As String
Dim NumInterventions As Integer
Dim NoScores As Integer
Dim Rating(5) As Integer
Dim NumGoals As Integer
Dim x As Integer
Dim tmpValue As Double
```

```
AssessmentText(0) = "MobilityAssessments"
AssessmentText(1) = "ILAssessment"
AssessmentText(2) = "SocialAssessment"
AssessmentText(3) = "EmploymentAssessment"
AssessmentText(4) = "ComputerAssessment"
```

```
InterventionID(0) = "014"      ' Mobility
InterventionID(1) = "013"      ' Independent Living
InterventionID(2) = "010"      ' Social Service
InterventionID(3) = "050"      ' Employment
InterventionID(4) = "160"      ' Computer Skills
```

```
' PreService - Self-Assessment Component - TaskA & TaskB from Assessments - A1
```

```
Set rsScores = New Recordset
```

```
While (x < 5)
    rsScores.Open "SELECT TaskA, TaskB FROM " & AssessmentText(x) & " WHERE ConsumerId = " & Cust
erID & " AND Episode = " & Episode & _
        " AND Agency = " & AgencyCode & " AND CaseNumber = " & CaseNumber & "'", cnr
    adOpenForwardOnly, adLockReadOnly
    With rsScores
        If Not (.EOF And .BOF) Then
            If ([taskA] <> "" And [taskB] <> "") Then
                PreSelf = PreSelf + CInt([taskA])
                PreProv = PreProv + CInt([taskB])
            Else
                NoScores = NoScores + 1 ' Assessments that have no scores, i.e. No service prov
ed
            End If
            .MoveNext
        End If
    End With

    x = x + 1
    rsScores.Close
Wend
```

```
' Total number of Interventions for the consumer
```

```
Set rsScores = New Recordset
rsScores.Open "SELECT count(InterventionID) as IntTotal FROM Interventions WHERE ConsumerId =
CustomerID & " AND Episode = " & Episode & _
    " AND Agency = " & AgencyCode & " AND CaseNumber = " & CaseNumber & "'", cnn,
penForwardOnly, adLockReadOnly
With rsScores
    If Not (.EOF And .BOF) Then
        NumInterventions = [IntTotal]
```

frmOverallScore - 2

60439708.0113

End If  
End With

rsScores.Close  
Set rsScores = Nothing

If NumInterventions > 5 Then ' Remove Non-Assessment Interventions  
NumInterventions = 5  
End If

NumInterventions = NumInterventions - NoScores ' Remove Interventions that have no Assessment  
ore

PreSelf = PreSelf / NumInterventions

' Social questions from End of Episode QA  
tmpValue = 0

tmpValue = frmEndOfEpisode.cboActivity.ItemData(frmEndOfEpisode.cboActivity.ListIndex)  
tmpValue = tmpValue + frmEndOfEpisode.cboFrequency2.ItemData(frmEndOfEpisode.cboFrequency2.Lis  
dex)  
tmpValue = tmpValue \* 12.5

PreSelf = (PreSelf + tmpValue) / 2

' Calculate Preservice Provider Component - AII  
PreProv = PreProv / NumInterventions  
PreServiceScore = (PreSelf + PreProv) / 2

txtPreScore = PreServiceScore

' Post Service - Self-Assessment Component - BI  
tmpValue = 0

tmpValue = frmEndOfEpisode.cboActivity.ItemData(frmEndOfEpisode.cboActivity.ListIndex)  
tmpValue = tmpValue + frmEndOfEpisode.cboFrequency2.ItemData(frmEndOfEpisode.cboFrequency2.Lis  
dex)  
tmpValue = tmpValue + frmEndOfEpisode.cboEffect.ItemData(frmEndOfEpisode.cboEffect.ListIndex)  
PostSelf = tmpValue \* 8.33

' Post Service - Provider's Determination of Functioning - BII  
NumGoals = 0  
tmpValue = 0

' Add the goal ratings for a particular intervention  
x = 0

While (x <> 5)  
Rating(x) = 0  
x = x + 1  
Wend

x = 0

While (x <> 5)  
Set rsScores = New Recordset  
rsScores.Open "SELECT Achievement FROM GoalAchieve WHERE ConsumerID = " & CustomerID & " AND  
isode = " & Episode & "  
AND Agency = " & AgencyCode & " AND CaseNumber = " & CaseNumber & "' & \_  
AND InterventionID = " & InterventionID(x) & "'", cnn, adOpenForwardOnly, ad  
kReadOnly  
With rsScores  
If Not (.EOF And .BOF) Then  
Rating(x) = Rating(x) + ![Achievement] ' Add up the ratings for each intervention (  
.rating(0) = Mobility, rating(1) = IL, etc.)  
NumGoals = NumGoals + 1 ' Total number of goals

mOverallScore - 3

60439708.01136

End If  
End With

x = x + 1  
Wend

rsScores.Close  
Set rsScores = Nothing

x = 0

While (x <> 5)  
Rating(x) = Rating(x) / 100  
Rating(x) = Rating(x) / (NumGoals \* 4)  
! number of goals \* 4)  
x = x + 1  
Wend

' Divide each intervention goal ratings by 100  
' Divide each intervention goal rating by (to

PostProv = Rating(0) + Rating(1) + Rating(2) + Rating(3) + Rating(4)

PostProv = PostProv / NumInterventions

PostServiceScore = (PostSelf + PostProv) / 2

txtPostScore = PostServiceScore

' Percent Change  
tmpValue = 0

tmpValue = PostServiceScore - PreServiceScore  
tmpValue = tmpValue / PreServiceScore  
tmpValue = tmpValue \* 100

txtFunctionalChange = tmpValue

End Sub

Private Sub Command1\_Click()  
Unload Me  
End Sub

```

Option Explicit
Dim rstPayer As ADODB.Recordset
Dim rstInterventions As ADODB.Recordset
Dim rstLocations As ADODB.Recordset
Dim rstStartDate As ADODB.Recordset
Dim rstAuth As ADODB.Recordset
Dim IndexNo As Integer
Dim InterventionSequence As Long
Dim PayerSequence As Long
Dim j As Integer
Dim k As Integer
Dim x As Integer
Dim IsNewPayer As Boolean
Dim IsNewAuth As Boolean
Dim SaveFlag As Boolean
Dim InterventionSeq(100) As Long ' Temporarily assign 100 until Ed finds out how many we need
Dim InterventionDesc(100) As String
Dim tmpID As String
Dim tmpAuthStartDate As Date
Dim tmpAuthEndDate As Date
Dim UnloadFlag As Boolean

Private Sub cboInterventioncodes_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboInterventionCodes.ListIndex = -1
    End If
End Sub

Private Sub cboInterventionCodes_LostFocus()
    If cboInterventionCodes.ListIndex <> -1 Then
        ' Find the Intervention Sequence number by the Description

        For x = 0 To 99
            If InterventionDesc(x) = cboInterventionCodes.Text Then
                InterventionSequence = InterventionSeq(x)
                Exit For
            End If
        Next

        ' Display the Start Date for the Intervention

        Set rstStartDate = New ADODB.Recordset
        rstStartDate.CursorLocation = adUseClient
        rstStartDate.Open "Select StartDate From Interventions" & _
            " Where ConsumerId = " & CustomerID & " And Agency = " & AgencyCode & _
            " And CaseNumber = " & CaseNumber & " And Episode = " & Episode & _
            " And InterventionSequence = " & InterventionSequence, cnn, adOpenDy
        c, adLockOptimistic
        With rstStartDate
            txtStartDate = ![StartDate]
        End With
        rstStartDate.Close
        Set rstStartDate = Nothing
    End If
End Sub

Private Sub cboPayerLocation_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboPayerLocation.ListIndex = -1
    End If
End Sub

Private Sub cboPayers_click()
    ' clear data in all fields
    If cboPayers.ListIndex = -1 Then

```

```
clearData cboPayerLocation, False
txtIDWithPayer = ""
txtPayerContact = ""
txtPayerPhone = ""
IsNewPayer = True
txtAuthTotal = ""
txtAuthorizationID = ""
clearData cboInterventionCodes, False
txtAuthStartDate = ""
txtAuthEndDate = ""
txtStartDate = ""
End If

If cboPayers.ListIndex <> -1 Then ' Display the locations for the payer that was selected
    Display_Location
End If

End Sub

Private Sub cboPayers_KeyUp(KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboPayers.ListIndex = -1
    End If
End Sub

Private Sub cmdClose_Click()
    Set rstPayer = Nothing
    Set rstAuth = Nothing
    Unload Me
End Sub

Private Sub cmdNewAuth_Click()
    ' Clear data in the Authorization fields
    txtAuthorizationID = ""
    clearData cboInterventionCodes, False
    txtStartDate = ""
    txtAuthStartDate = ""
    txtAuthEndDate = ""

    ' Disable all navigation buttons except Save Payer and Intervention
    cmdPreviousAuth.Enabled = False
    cmdNextAuth.Enabled = False
    cmdNewAuth.Enabled = False
    cmdPreviousPayer.Enabled = False
    cmdNextPayer.Enabled = False
    cmdNewPayer.Enabled = False
    cboInterventionCodes.Locked = False

    txtAuthorizationID.SetFocus

    IsNewAuth = True

End Sub

Private Sub cmdNewPayer_Click()
    ' Clear all data in all fields
    clearData cboPayers, False
    clearData cboPayerLocation, False
    txtIDWithPayer = ""
    txtPayerContact = ""
    txtPayerPhone = ""
    IsNewPayer = True
    txtAuthTotal = ""
    txtAuthorizationID = ""
    clearData cboInterventionCodes, False
    txtAuthStartDate = ""
    txtAuthEndDate = ""
    txtStartDate = ""
```

```

' Disable all navigation buttons except Save Payer
cmdPreviousAuth.Enabled = False
cmdNextAuth.Enabled = False
cmdNewAuth.Enabled = False
cmdPreviousPayer.Enabled = False
cmdNextPayer.Enabled = False
cmdNewPayer.Enabled = False
cboInterventionCodes.Locked = False
' Enable data fields
cboPayers.Enabled = True
cboPayerLocation.Enabled = True
txtIDWithPayer.Enabled = True
txtPayerContact.Enabled = True
txtPayerPhone.Enabled = True
txtAuthorizationID.Enabled = True
cboInterventionCodes.Enabled = True

cboPayers.SetFocus

End Sub

Private Sub cmdNextAuth_Click()

SaveRecords

IsNewAuth = True

With rstAuth
    If Not (.BOF And .EOF) Then
        .MoveNext
        txtAuthorizationID = ![AuthorizationID]
        txtAuthTotal = .RecordCount
        LoadDataToControls rstAuth, txtAuthStartDate, False
        LoadDataToControls rstAuth, txtAuthEndDate, False
        InterventionSequence = ![InterventionSequence]
        IsNewAuth = False
    End If

    x = 0
    ' Highlight the saved Intervention in the combo box
    For x = 0 To 99
        If InterventionSequence = InterventionSeq(x) Then
            cboInterventionCodes.ListIndex = x
        End If
    Next

    ' Refresh the StartDate for the Intervention

    If InterventionSequence <> 0 Then
        Set rstStartDate = New ADODB.Recordset
        rstStartDate.CursorLocation = adUseClient

        rstStartDate.Open "Select StartDate From Interventions" & _
            " Where ConsumerId = " & CustomerID & " And Agency = " & AgencyCode & _
            " And CaseNumber = " & CaseNumber & " And Episode = " & Episode & _
            " And InterventionSequence = " & InterventionSequence, cnn, adOpenDy

c, adLockOptimistic
        With rstStartDate
            txtStartDate = ![StartDate]
        End With
        rstStartDate.Close
    End If

    ' Enable/Disable authorization navigation buttons accordingly

    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then
            cmdPreviousAuth.Enabled = False

```



```
cmdNextAuth.Enabled = True
cmdNewAuth.Enabled = False
ElseIf .AbsolutePosition < .RecordCount Then
cmdPreviousAuth.Enabled = True
cmdNextAuth.Enabled = True
cmdNewAuth.Enabled = False
ElseIf .RecordCount = .AbsolutePosition Then
cmdPreviousAuth.Enabled = True
cmdNextAuth.Enabled = False
cmdNewAuth.Enabled = True
End If
End If
End With

txtAuthorizationID.SetFocus

End Sub

Private Sub cmdNextPayer_Click()

SaveRecords

With rstPayer
.MoveNext
PayerSequence = ![PayerSequence]

LoadDataToControls rstPayer, txtPayerContact, False
LoadDataToControls rstPayer, txtPayerPhone, False
LoadDataToControls rstPayer, txtIDWithPayer, False
LoadDataToControls rstPayer, cboPayers, False

' Enable and disable navigation buttons accordingly

If .RecordCount >= 1 Then
If .AbsolutePosition = 1 Then
cmdPreviousPayer.Enabled = False
cmdNextPayer.Enabled = True
cmdNewPayer.Enabled = False
ElseIf .AbsolutePosition < .RecordCount Then
cmdPreviousPayer.Enabled = True
cmdNextPayer.Enabled = True
cmdNewPayer.Enabled = False
ElseIf .RecordCount = .AbsolutePosition Then
cmdPreviousPayer.Enabled = True
cmdNextPayer.Enabled = False
cmdNewPayer.Enabled = True
End If
End If

End With

' Load the location codes and descriptions for the payer that was saved

cboPayerLocation.Clear

Set rstLocations = New ADODB.Recordset
rstLocations.CursorLocation = adUseClient
rstLocations.Open "Select Code, Description From PayerLocation" & _
" Where PayerID = '" & cboPayers.ItemData(cboPayers.ListIndex) & "'", cnn,
enDynamic, adLockOptimistic

With rstLocations
.MoveFirst
While Not (.EOF)
cboPayerLocation.AddItem ![Description]
cboPayerLocation.ItemData(cboPayerLocation.NewIndex) = ![Code]
.MoveNext
Wend
End With
```

```
rstLocations.Close
```

```
' Display the Location that was saved
```

```
Set rstLocations = New ADODB.Recordset
rstLocations.CursorLocation = adUseClient
rstLocations.Open "Select PayerLocation From PayerInfo" & _
    " Where ConsumerID = " & CustomerID & " And Agency = " & AgencyCode & " " & _
    " And CaseNumber = " & CaseNumber & " And Episode = " & Episode & _
    " And PayerID = " & cboPayers.ItemData(cboPayers.ListIndex) & " ", cnn,
    adOpenDynamic, adLockOptimistic
```

```
With rstLocations
    .MoveFirst
```

```
    While Not (.EOF)
        If Trim(![PayerLocation] & vbNullString) = "" Then
            cboPayerLocation.ListIndex = -1
        Else
            For j = 0 To cboPayerLocation.ListCount - 1
                If cboPayerLocation.ItemData(j) = ![PayerLocation] Then
                    k = j
                    Exit For
                End If
            Next
            cboPayerLocation.ListIndex = k
        End If
        .MoveNext
    Wend
End With
```

```
rstLocations.Close
```

```
' Authorization Section
```

```
IsNewAuth = True
```

```
Set rstAuth = New ADODB.Recordset
rstAuth.CursorLocation = adUseClient
rstAuth.Open " Select * From PayerAuth Where ConsumerID = " & CustomerID & " And Agency = " & _
    AgencyCode & " " & CaseNumber = " & CaseNumber & " And Episode = " & Episode & _
    " And PayerSequence = " & PayerSequence & " Order By PayerAuthSequence", cnn, ad
    OpenDynamic, adLockOptimistic
```

```
With rstAuth
```

```
    If Not (.BOF And .EOF) Then
        .MoveFirst
        txtAuthorizationID = ![AuthorizationID]
        txtAuthTotal = .RecordCount
        LoadDataToControls rstAuth, txtAuthStartDate, False
        LoadDataToControls rstAuth, txtAuthEndDate, False
        InterventionSequence = ![InterventionSequence]
        IsNewAuth = False
    End If
```

```
x = 0 ' Highlight the saved Intervention in the combo box
```

```
For x = 0 To 99
    If InterventionSequence = InterventionSeq(x) Then
        cboInterventionCodes.ListIndex = x
    End If
Next
```

```
' Refresh the StartDate for the Intervention
```

```
If InterventionSequence <> 0 Then
    Set rstStartDate = New ADODB.Recordset
```

```

rstStartDate.CursorLocation = adUseClient

rstStartDate.Open "Select StartDate From Interventions" & _
    " Where ConsumerId = " & CustomerID & " And Agency = '" & AgencyCode & "'
    & _
    " And CaseNumber = '" & CaseNumber & "' And Episode = '" & Episode & _
    " And InterventionSequence = '" & InterventionSequence, cnn, adOpenDynam

c, adLockOptimistic
With rstStartDate
    txtStartDate = ![StartDate]
End With
rstStartDate.Close
End If

' Enable/Disable authorization buttons according to how many records we found in the records
' Remember we are always looking at the first authorization for the payer

If .RecordCount > 0 Then
    If .AbsolutePosition = 1 And .RecordCount = 1 Then ' just one record in the recordset
        cmdPreviousAuth.Enabled = False
        cmdNextAuth.Enabled = False
        cmdNewAuth.Enabled = True
    ElseIf .AbsolutePosition = 1 And .RecordCount > 1 Then ' More than one record in the r
ordset
        cmdPreviousAuth.Enabled = False
        cmdNextAuth.Enabled = True
        cmdNewAuth.Enabled = False
    ElseIf .AbsolutePosition > 1 And .AbsolutePosition = .RecordCount Then ' At the end of
e recordset
        cmdPreviousAuth.Enabled = True
        cmdNextAuth.Enabled = False
        cmdNewAuth.Enabled = True
    End If
End If

End With

cboPayers.SetFocus

End Sub

Private Sub cmdPreviousAuth_Click()

SaveRecords

IsNewAuth = True

With rstAuth
    If Not (.BOF And .EOF) Then
        .MovePrevious
        txtAuthorizationID = ![AuthorizationID]
        txtAuthTotal = .RecordCount
        LoadDataToControls rstAuth, txtAuthStartDate, False
        LoadDataToControls rstAuth, txtAuthEndDate, False
        InterventionSequence = ![InterventionSequence]
        IsNewAuth = False
    End If

    x = 0 ' Highlight the saved Intervention in the combo box
    For x = 0 To 99
        If InterventionSequence = InterventionSeq(x) Then
            cboInterventionCodes.ListIndex = x
        End If
    Next

    ' Refresh the StartDate for the Intervention

    If InterventionSequence <> 0 Then

```

```
Set rstStartDate = New ADODB.Recordset
rstStartDate.CursorLocation = adUseClient
```

```
rstStartDate.Open "Select StartDate From Interventions" & _
    " Where ConsumerId = " & CustomerID & " And Agency = " & AgencyCode & "
    & _
    " And CaseNumber = " & CaseNumber & " And Episode = " & Episode & _
    " And InterventionSequence = " & InterventionSequence, cnn, adOpenForwa
```

```
dOnly, adLockReadOnly
With rstStartDate
    txtStartDate = ![StartDate]
End With
rstStartDate.Close
End If
```

' Enable/Disable authorization navigation buttons accordingly

```
If .RecordCount = 1 And .AbsolutePosition = 1 Then
    cmdPreviousAuth.Enabled = False
    cmdNextAuth.Enabled = True
    cmdNewAuth.Enabled = True
End If
```

```
If .RecordCount > 1 Then
    If .AbsolutePosition = 1 Then
        cmdPreviousAuth.Enabled = False
        cmdNextAuth.Enabled = True
        cmdNewAuth.Enabled = False
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPreviousAuth.Enabled = True
        cmdNextAuth.Enabled = True
        cmdNewAuth.Enabled = False
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousAuth.Enabled = True
        cmdNextAuth.Enabled = False
        cmdNewAuth.Enabled = True
    End If
End If
End With
```

```
txtAuthorizationID.SetFocus
```

```
End Sub
```

```
Private Sub cmdPreviousPayer_Click()
```

```
SaveRecords
```

```
With rstPayer
    .MovePrevious
    PayerSequence = ![PayerSequence]
```

```
LoadDataToControls rstPayer, txtPayerContact, False
LoadDataToControls rstPayer, txtPayerPhone, False
LoadDataToControls rstPayer, txtIDWithPayer, False
LoadDataToControls rstPayer, cboPayers, False
```

' Enable and disable navigation buttons accordingly

```
If .RecordCount >= 1 Then
    If .AbsolutePosition = 1 Then
        cmdPreviousPayer.Enabled = False
        cmdNextPayer.Enabled = True
        cmdNewPayer.Enabled = False
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPreviousPayer.Enabled = True
        cmdNextPayer.Enabled = True
        cmdNewPayer.Enabled = False
    ElseIf .RecordCount = .AbsolutePosition Then
```

```

cmdPreviousPayer.Enabled = True
cmdNextPayer.Enabled = False
cmdNewPayer.Enabled = True
End If
End If

End With

' Load the location codes and descriptions for the payer that was saved
cboPayerLocation.Clear

Set rstLocations = New ADODB.Recordset
rstLocations.CursorLocation = adUseClient
rstLocations.Open "Select Code, Description From PayerLocation" & _
                  " Where PayerID = '" & cboPayers.ItemData(cboPayers.ListIndex) & "'", cnn, _
                  adDynamic, adLockOptimistic

With rstLocations
    .MoveFirst
    While Not (.EOF)
        cboPayerLocation.AddItem ![Description]
        cboPayerLocation.ItemData(cboPayerLocation.NewIndex) = ![Code]
    .MoveNext
    Wend
End With

rstLocations.Close

' Display the Location that was saved

Set rstLocations = New ADODB.Recordset
rstLocations.CursorLocation = adUseClient
rstLocations.Open "Select PayerLocation From PayerInfo" & _
                  " Where ConsumerID = '" & CustomerID & "' And Agency = '" & AgencyCode & "' _
                  " And CaseNumber = '" & CaseNumber & "' And Episode = '" & Episode & "' _
                  " And PayerID = '" & cboPayers.ItemData(cboPayers.ListIndex) & "'", cnn, _
                  adDynamic, adLockOptimistic

With rstLocations
    .MoveFirst

    While Not (.EOF)
        If Trim(![PayerLocation] & vbNullString) = "" Then
            cboPayerLocation.ListIndex = -1
        Else
            For j = 0 To cboPayerLocation.ListCount - 1
                If cboPayerLocation.ItemData(j) = ![PayerLocation] Then
                    k = j
                    Exit For
                End If
            Next
            cboPayerLocation.ListIndex = k
        End If
        .MoveNext
    Wend
End With

rstLocations.Close

' Authorization Section

IsNewAuth = True

Set rstAuth = New ADODB.Recordset
rstAuth.CursorLocation = adUseClient
rstAuth.Open " Select * From PayerAuth Where ConsumerID = '" & CustomerID & "' And Agency = '" & _
            AgencyCode & "' And CaseNumber = '" & CaseNumber & "' And Episode = '" & Episode & "' _
            " And PayerID = '" & cboPayers.ItemData(cboPayers.ListIndex) & "'", cnn, _
            adDynamic, adLockOptimistic

```

```

    "And PayerSequence = " & PayerSequence & " Order By PayerAuthSequence", cnn, ad
nDynamic, adLockOptimistic

With rstAuth
    If Not (.BOF And .EOF) Then
        .MoveFirst
        txtAuthorizationID = ![AuthorizationID]
        txtAuthTotal = .RecordCount
        LoadDataToControls rstAuth, txtAuthStartDate, False
        LoadDataToControls rstAuth, txtAuthEndDate, False
        InterventionSequence = ![InterventionSequence]
        IsNewAuth = False
    End If

    x = 0                ' Highlight the saved Intervention in the combo box

    For x = 0 To 99
        If InterventionSequence = InterventionSeq(x) Then
            cboInterventionCodes.ListIndex = x
        End If
    Next

    ' Refresh the StartDate for the Intervention

    If InterventionSequence <> 0 Then
        Set rstStartDate = New ADODB.Recordset
        rstStartDate.CursorLocation = adUseClient

        rstStartDate.Open "Select StartDate From Interventions" & _
            " Where ConsumerId = " & CustomerID & " And Agency = " & AgencyCode &
            " And CaseNumber = " & CaseNumber & " And Episode = " & Episode &
            " And InterventionSequence = " & InterventionSequence, cnn, adOpenDy

c, adLockOptimistic
        With rstStartDate
            txtStartDate = ![StartDate]
        End With
        rstStartDate.Close
    End If

    ' Enable/Disable authorization navigation buttons accordingly

    If .RecordCount > 0 Then
        If .AbsolutePosition = 1 And .RecordCount = 1 Then    ' just one record in the recordse
            cmdPreviousAuth.Enabled = False
            cmdNextAuth.Enabled = False
            cmdNewAuth.Enabled = True
        ElseIf .AbsolutePosition = 1 And .RecordCount > 1 Then    ' More than one record in the
ordset
            cmdPreviousAuth.Enabled = False
            cmdNextAuth.Enabled = True
            cmdNewAuth.Enabled = False
        ElseIf .AbsolutePosition > 1 And .AbsolutePosition = .RecordCount Then    ' At the end c
e recordset
            cmdPreviousAuth.Enabled = True
            cmdNextAuth.Enabled = False
            cmdNewAuth.Enabled = True
        End If
    End If

End With

cboPayers.SetFocus

End Sub

Private Sub cmdSave_Click()

    SaveRecords

```

```
If SaveFlag = True Then
    MsgBox "Payer Information successfully saved!", vbOKOnly, "Save"
End If
```

```
End Sub
```

```
Private Sub SaveRecords()
```

```
Dim tmpPayerID As String
Dim tmpLocation As String
Dim tmpIDWithPayer As String
Dim tmpContact As String
Dim tmpPhone As String
Dim tmpFinishDate As String
Dim tmpPosition As Long
```

```
On Error GoTo ErrorSave
```

```
SaveFlag = True
```

```
' Check all mandatory fields have values
```

```
If cboPayers.ListIndex = -1 Then
    MsgBox "Please select a Payer!", vbInformation + vbOKOnly, "Save"
    cboPayers.SetFocus
    SaveFlag = False
    Exit Sub
```

```
Else
    tmpPayerID = cboPayers.ItemData(cboPayers.ListIndex)
End If
```

```
If cboPayerLocation.ListIndex = -1 Then
    MsgBox "Please select a location!", vbInformation + vbOKOnly, "Payer"
    cboPayerLocation.SetFocus
    SaveFlag = False
    Exit Sub
```

```
Else
    tmpLocation = cboPayerLocation.ItemData(cboPayerLocation.ListIndex)
End If
```

```
If Trim(txtIDWithPayer) = "" Then
    tmpIDWithPayer = ""
Else
    tmpIDWithPayer = txtIDWithPayer
End If
```

```
If Trim(txtPayerContact) = "" Then
    tmpContact = ""
Else
    tmpContact = txtPayerContact
End If
```

```
If Trim(txtPayerPhone) = "" Then
    tmpPhone = ""
Else
    tmpPhone = txtPayerPhone
End If
```

```
If Trim(txtAuthorizationID & vbNullString) = "" Then
    MsgBox "Please enter a Authorization Number!", vbOKOnly, "Save"
    txtAuthorizationID.SetFocus
    SaveFlag = False
    Exit Sub
End If
```

```
If cboInterventionCodes.ListIndex = -1 Then
    MsgBox "Please select an Intervention!", vbInformation + vbOKOnly, "Save"
    cboInterventionCodes.SetFocus
    SaveFlag = False
    Exit Sub
```

End If

```
If Not IsDate(Trim(txtAuthStartDate)) Then
    tmpAuthStartDate = "12:00:00 PM"
Else
    tmpAuthStartDate = txtAuthStartDate
End If
```

```
If Not IsDate(Trim(txtAuthEndDate)) Then
    tmpAuthEndDate = "12:00:00 PM"
Else
    tmpAuthEndDate = txtAuthEndDate
End If
```

If IsNewPayer = False Then     ' The user updated a Payer record

```
With rstPayer.
    ![PayerID] = tmpPayerID
    ![PayerLocation] = tmpLocation
    ![IDWithPayer] = tmpIDWithPayer
    ![PayerContact] = tmpContact
    ![PayerPhone] = tmpPhone
    .Update
End With
```

PayerSequence = rstPayer![PayerSequence]

Else     ' Inserting a record for the consumer

```
With rstPayer
    .AddNew
    ![consumerid] = CustomerID
    ![Agency] = AgencyCode
    ![CaseNumber] = CaseNumber
    ![Episode] = Episode
    ![PayerID] = tmpPayerID
    ![PayerLocation] = tmpLocation
    ![IDWithPayer] = tmpIDWithPayer
    ![PayerContact] = tmpContact
    ![PayerPhone] = tmpPhone
    .Update
End With
```

```
rstPayer.Close
rstPayer.Open " Select * From PayerInfo Where 'ConsumerID = " & CustomerID & " And Agency = " &
AgencyCode & " And CaseNumber = " & CaseNumber & " And Episode = " & Episode, cnn, adOpenDynamic, adLockOptimist
rstPayer.MoveLast
txtTotal = rstPayer.RecordCount
```

PayerSequence = rstPayer![PayerSequence]

With rstPayer

```
If .RecordCount > 1 Then
    If .AbsolutePosition = 1 Then
        cmdPreviousPayer.Enabled = False
        cmdNextPayer.Enabled = True
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPreviousPayer.Enabled = True
        cmdNextPayer.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousPayer.Enabled = True
        cmdNextPayer.Enabled = False
    End If
End If
```

```
If .RecordCount = 1 And .AbsolutePosition = 1 Then
    cmdPreviousPayer.Enabled = False
```



```
cmdNextPayer.Enabled = False
cmdNewPayer.Enabled = True
End If
End With

IsNewPayer = False
IsNewAuth = True

End If

cboPayers.SetFocus

' Update or Insert data to the Authorizations table
If IsNewAuth = False Then      ' Update only

    With rstAuth
        ![PayerSequence] = PayerSequence
        ![InterventionSequence] = InterventionSequence
        ![AuthorizationID] = txtAuthorizationID
        ![AuthStartDate] = tmpAuthStartDate
        ![AuthEndDate] = tmpAuthEndDate
        .Update
    End With

Else                            ' Insert new record

    With rstAuth
        .AddNew
        ![consumerid] = CustomerID
        ![Agency] = AgencyCode
        ![CaseNumber] = CaseNumber
        ![Episode] = Episode
        ![PayerSequence] = PayerSequence
        ![InterventionSequence] = InterventionSequence
        ![AuthorizationID] = txtAuthorizationID
        ![AuthStartDate] = tmpAuthStartDate
        ![AuthEndDate] = tmpAuthEndDate
        .Update
    End With

    rstAuth.Close
    rstAuth.Open " Select * From PayerAuth Where ConsumerID = " & CustomerID & " And Agency = " &
AgencyCode & " And CaseNumber = " & CaseNumber & " And Episode = " & Episode & _
        "And PayerSequence = " & PayerSequence, cnn, adOpenDynamic, adLockOptimistic
    rstAuth.MoveLast
    txtAuthTotal = rstAuth.RecordCount
    InterventionSequence = rstAuth![InterventionSequence]

    With rstAuth
        If .RecordCount > 0 Then
            If .AbsolutePosition = 1 And .RecordCount = 1 Then
                cmdPreviousAuth.Enabled = False
                cmdNextAuth.Enabled = False
                cmdNewPayer.Enabled = True
            ElseIf .AbsolutePosition < .RecordCount Then
                cmdPreviousAuth.Enabled = True
                cmdNextAuth.Enabled = True
            ElseIf .RecordCount = .AbsolutePosition Then
                cmdPreviousAuth.Enabled = True
                cmdNextAuth.Enabled = False
            End If
        End If
    End With

    IsNewAuth = False
    txtAuthorizationID.SetFocus
```

```
cmdNewAuth.Enabled = True

' enable Payer navigation buttons accordingly
With rstPayer
    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then
            cmdPreviousPayer.Enabled = False
            cmdNextPayer.Enabled = True
            cmdNewPayer.Enabled = False
        ElseIf .AbsolutePosition < .RecordCount Then
            cmdPreviousPayer.Enabled = True
            cmdNextPayer.Enabled = True
            cmdNewPayer.Enabled = False
        ElseIf .RecordCount = .AbsolutePosition Then
            cmdPreviousPayer.Enabled = True
            cmdNextPayer.Enabled = False
            cmdNewPayer.Enabled = True
        End If
    End If
End With

End If

Exit Sub

ErrorSave:
MsgBox Err.Description, vbCritical + vbOKCancel, "Save"
End Sub

Private Sub Display_Location()
    ' Load the location codes and descriptions for the payer that was saved

    cboPayerLocation.Clear

    Set rstLocations = New ADODB.Recordset
    rstLocations.CursorLocation = adUseClient
    rstLocations.Open "Select Code, Description From PayerLocation" & _
        " Where PayerID = " & cboPayers.ItemData(cboPayers.ListIndex) & "'", cnn, ad
    enDynamic, adLockOptimistic

    With rstLocations
        .MoveFirst
        While Not (.EOF)
            cboPayerLocation.AddItem ![Description]
            cboPayerLocation.ItemData(cboPayerLocation.NewIndex) = ![Code]
            .MoveNext
        Wend
    End With

    rstLocations.Close
    Set rstLocations = Nothing

    cboPayerLocation.ListIndex = 0      ' Set focus to the first item as a default

End Sub

Private Sub Form_Load()
    ' Consumer Information
    txtConsumerName.Text = FirstName & " " & LastName
    txtCaseNumber = CaseNumber
    txtEpisode = Episode

    ' Load codes and descriptions for Payers
    LoadControlsItems cboPayers, False

    ' Load Intervention codes and descriptions for this consumer into the combo box
    IndexNo = 0
```

```
cboInterventionCodes.Clear
```

```
Set rstInterventions = New ADODB.Recordset
rstInterventions.CursorLocation = adUseClient
rstInterventions.Open "Select Description, StartDate, InterventionSequence From InterventionCo
IC, Interventions I " & _
    "Where IC.Code = I.InterventionID " & _
    " And I.ConsumerID = " & CustomerID & " And I.Agency = " & AgencyCode &
    " & _
    " And I.CaseNumber = " & CaseNumber & " " & _
    " And I.Episode = " & Episode, cnn, adOpenDynamic, adLockOptimistic
```

```
With rstInterventions
    If rstInterventions.BOF = False And rstInterventions.EOF = False Then
        .MoveFirst
        While Not (.EOF)
            cboInterventionCodes.AddItem ![Description]
            cboInterventionCodes.ItemData(cboInterventionCodes.NewIndex) = IndexNo
            InterventionSeq(IndexNo) = ![InterventionSequence]
            InterventionDesc(IndexNo) = ![Description]
            .MoveNext
            IndexNo = IndexNo + 1
        Wend
    End If
End With
```

```
CloseRst:
rstInterventions.Close
```

```
' Load Payers from the database
```

```
IsNewPayer = False
```

```
Set rstPayer = New ADODB.Recordset
rstPayer.CursorLocation = adUseClient
rstPayer.Open " Select * From PayerInfo Where ConsumerID = " & CustomerID & " And Agency = " &
AgencyCode & " And CaseNumber = " & CaseNumber & " And Episode = " & Episode & " Order By PayerID, PayerSequence,
erLocation", cnn, adOpenDynamic, adLockOptimistic
```

```
If rstPayer.RecordCount = 0 Then
    IsNewPayer = True
    GoTo Authorizations
End If
```

```
With rstPayer
    If Not (.BOF And .EOF) Then
        .MoveFirst

        PayerSequence = ![PayerSequence]
        txtTotal = .RecordCount

        LoadDataToControls rstPayer, txtPayerContact, False
        LoadDataToControls rstPayer, txtPayerPhone, False
        LoadDataToControls rstPayer, txtIDWithPayer, False
        LoadDataToControls rstPayer, cboPayers, False

        If .RecordCount = 1 And .AbsolutePosition = 1 Then
            cmdPreviousPayer.Enabled = False
            cmdNextPayer.Enabled = False
            cmdNewPayer.Enabled = True
        End If

        If .RecordCount > 1 Then
            If .AbsolutePosition = 1 Then
                cmdPreviousPayer.Enabled = False
                cmdNextPayer.Enabled = True
                cmdNewPayer.Enabled = False
            ElseIf .AbsolutePosition < .RecordCount Then
```

```

        cmdPreviousPayer.Enabled = True
        cmdNextPayer.Enabled = True
        cmdNewPayer.Enabled = False
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousPayer.Enabled = True
        cmdNextPayer.Enabled = False
        cmdNewPayer.Enabled = True
    End If
End If
End If
End With

Display_Location

' Load the location codes and descriptions for the payer that was saved

' cboPayerLocation.Clear

'Set rstLocations = New ADODB.Recordset
rstLocations.CursorLocation = adUseClient
rstLocations.Open "Select Code, Description From PayerLocation" & _
    " Where PayerID = '" & cboPayers.ItemData(cboPayers.ListIndex) & "'", cnn, a
penDynamic, adLockOptimistic

'With rstLocations
' .MoveFirst
' While Not (.EOF)
'     cboPayerLocation.AddItem ![Description]
'     cboPayerLocation.ItemData(cboPayerLocation.NewIndex) = ![Code]
'     .MoveNext
' Wend
'End With

rstLocations.Close
Set rstLocations = Nothing

' Display the Location that was saved

Set rstLocations = New ADODB.Recordset
rstLocations.CursorLocation = adUseClient
rstLocations.Open "Select PayerLocation From PayerInfo" & _
    " Where ConsumerID = '" & CustomerID & "' And Agency = '" & AgencyCode & "'
    " And CaseNumber = '" & CaseNumber & "' And Episode = '" & Episode & _
    " And PayerID = '" & cboPayers.ItemData(cboPayers.ListIndex) & "'", cnn,
penDynamic, adLockOptimistic

With rstLocations
    .MoveFirst

    While Not (.EOF)
        If Trim(![PayerLocation] & vbNullString) = "" Then
            cboPayerLocation.ListIndex = -1
        Else
            For j = 0 To cboPayerLocation.ListCount - 1
                If cboPayerLocation.ItemData(j) = ![PayerLocation] Then
                    k = j
                    Exit For
                End If
            Next
            cboPayerLocation.ListIndex = k
        End If
        .MoveNext
    Wend
End With

rstLocations.Close
Set rstLocations = Nothing

Authorizations:

```

```
' Authorizations Section
' Load Authorizations from the database
```

```
IsNewAuth = True
```

```
Set rstAuth = New ADODB.Recordset
rstAuth.CursorLocation = adUseClient
rstAuth.Open " Select * From PayerAuth Where ConsumerID = " & CustomerID & " And Agency = " & AgencyCode & " And CaseNumber = " & CaseNumber & " And Episode = " & Episode & " And PayerSequence = " & PayerSequence, cnn, adOpenDynamic, adLockOptimistic
```

```
If rstAuth.RecordCount = 0 Then
    cmdPreviousAuth.Enabled = False
    cmdNextAuth.Enabled = False
    cmdNewAuth.Enabled = False
    GoTo Finishup
End If
```

```
With rstAuth
```

```
    If Not (.BOF And .EOF) Then
```

```
        .MoveFirst
```

```
        txtAuthorizationID = ![AuthorizationID]
```

```
        txtAuthTotal = .RecordCount
```

```
        LoadDataToControls rstAuth, txtAuthStartDate, False
```

```
        LoadDataToControls rstAuth, txtAuthEndDate, False
```

```
        InterventionSequence = ![InterventionSequence]
```

```
        IsNewAuth = False
```

```
        Set rstStartDate = New ADODB.Recordset
```

```
        rstStartDate.CursorLocation = adUseClient
```

```
        rstStartDate.Open "Select StartDate From Interventions" & _
            " Where ConsumerId = " & CustomerID & " And Agency = " & AgencyCode &
```

```
            " & _
            " And CaseNumber = " & CaseNumber & " And Episode = " & Episode & _
            " And InterventionSequence = " & InterventionSequence, cnn, adOpenDyr
```

```
ic, adLockOptimistic
```

```
    With rstStartDate
```

```
        txtStartDate = ![StartDate]
```

```
    End With
```

```
    rstStartDate.Close
```

```
    Set rstStartDate = Nothing
```

```
x = 0
```

```
' Highlight the saved Intervention in the combo box
```

```
For x = 0 To 99
```

```
    If InterventionSequence = InterventionSeq(x) Then
```

```
        cboInterventionCodes.ListIndex = x
```

```
    End If
```

```
Next
```

```
' Enable/Disable authorization navigation buttons accordingly
```

```
If .RecordCount = 1 And .AbsolutePosition = 1 Then
```

```
    cmdPreviousAuth.Enabled = False
```

```
    cmdNextAuth.Enabled = False
```

```
    cmdNewAuth.Enabled = True
```

```
End If
```

```
If .RecordCount > 1 Then
```

```
    If .AbsolutePosition = 1 Then
```

```
        cmdPreviousAuth.Enabled = False
```

```
        cmdNextAuth.Enabled = True
```

```
    ElseIf .AbsolutePosition < .RecordCount Then
```

```
        cmdPreviousAuth.Enabled = True
```

```
        cmdNextAuth.Enabled = True
```

```
    ElseIf .RecordCount = .AbsolutePosition Then
```

```
        cmdPreviousAuth.Enabled = True
```

```
        cmdNextAuth.Enabled = False
```

```
    End If
```

```
End If
End If
End With

Finishup:
Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"
End Sub

Private Sub Form_Unload(Cancel As Integer)

Here:
    Set rstPayer = Nothing
    Set rstAuth = Nothing
    Unload Me
End Sub

Private Sub txtPayerPhone_Change()
    If Len(txtPayerPhone) = 4 Then
        If Not IsNumeric(Mid(txtPayerPhone, 2, 3)) Then
            MsgBox "Please enter a numeric number for the phone number!", vbInformation + vbOKOnly
            " Phone Number"
            txtPayerPhone = Mid(txtPayerPhone, 1, 2)
            txtPayerPhone.SelStart = 2
        Else
            txtPayerPhone = txtPayerPhone & "-"
            txtPayerPhone.SelStart = 7
        End If
    ElseIf Len(txtPayerPhone) = 9 Then
        If Not IsNumeric(Mid(txtPayerPhone, 7, 3)) Then
            MsgBox "Please enter a numeric number for the phone number!", vbInformation + vbOKOnly
            " Phone Number"
            txtPayerPhone = Mid(txtPayerPhone, 1, 6)
            txtPayerPhone.SelStart = 7
        Else
            txtPayerPhone = txtPayerPhone & "--"
            txtPayerPhone.SelStart = 11
        End If
    ElseIf Len(txtPayerPhone) = 14 Then
        If Not IsNumeric(Mid(txtPayerPhone, 11, 4)) Then
            MsgBox "Please enter a numeric number for the phone number!", vbInformation + vbOKOnly
            " Phone Number"
            txtPayerPhone = Mid(txtPayerPhone, 1, 10)
            txtPayerPhone.SetFocus
        End If
    End If
End Sub

Private Sub txtPayerPhone_GotFocus()
    If txtPayerPhone = "" Then
        txtPayerPhone = "("
        txtPayerPhone.SelStart = 2
    End If
End Sub

Private Sub txtPayerPhone_KeyUp(KeyCode As Integer, Shift As Integer)
    If txtPayerPhone = "" Or KeyCode = vbKeyBack Then
        txtPayerPhone = "("
        txtPayerPhone.SelStart = 2
    End If
End Sub

Private Sub txtPayerPhone_LostFocus()
    If txtPayerPhone = "(" Then txtPayerPhone = ""
End Sub
```

Option Explicit

Dim AgencyName As String

Dim rsQA As New Recordset

Private Sub cboVisionChanged\_LostFocus()

    If cboVisionChanged.ListIndex = 1 Then  
        cboActivity.SetFocus

    End If

End Sub

Private Sub cmdClose\_Click()

    Unload Me

End Sub

Private Sub cmdSave\_Click()

    SaveRecords

End Sub

Private Sub SaveRecords()

    Dim tmpBuf1 As String

    Dim j As Integer

    Dim tmpParticipantNumber As String

    Dim tmpInterviewDate2 As Date

    Dim tmpStaff As String

    Dim tmpPositive As String

    Dim tmpDPositive As String                   Areas(0)

    Dim tmpPositiveDescription As String

    Dim tmpNewWays As String

    Dim tmpNewTechniques As String           Frequency1

    Dim tmpNoTechniques As String

    Dim tmpCantDo As String

    Dim tmpAreas As String

    Dim tmpCDescription As String

    Dim tmpVisionChanged As String

    Dim tmpHowChanged As String

    Dim tmpMakeAppointment As String

    Dim tmpPostActivity As String

    Dim tmpPostSocial As String

    Dim tmpPostFunctioning As String

    Dim tmpEndDate2 As Date

On Error GoTo errSave

If Trim(txtParticipantNumber) = "" Then

    MsgBox "You must enter a Participant Number!", vbOKOnly, "Save"

    txtParticipantNumber.SetFocus

    Exit Sub

Else

    tmpParticipantNumber = txtParticipantNumber

End If

If Trim(txtInterviewDate2) = "" Then

    MsgBox "You must enter an Interview Date!", vbOKOnly, "Save"

    txtInterviewDate2.SetFocus

    Exit Sub

Else

    tmpInterviewDate2 = txtInterviewDate2

End If

If cboStaffs.ListIndex = -1 Then

    MsgBox "You must enter a Staff ID!", vbOKOnly, "Save"

    cboStaffs.SetFocus

    Exit Sub

Else

    tmpStaff = cboStaffs.ItemData(cboStaffs.ListIndex)

End If

If cboPositive.ListIndex = -1 Then

    tmpPositive = ""

Else

```
tmpPositive = cboPositive.ItemData(cboPositive.ListIndex)
End If

For j = 0 To lstAreas(0).ListCount - 1
    If lstAreas(0).Selected(j) Then
        tmpBuf1 = tmpBuf1 & "1"
    Else
        tmpBuf1 = tmpBuf1 & "0"
    End If
Next

tmpDPositive = tmpBuf1
tmpBuf1 = ""

If Trim(txtPositiveDescription) = "" Then
    tmpPositiveDescription = ""
Else
    tmpPositiveDescription = txtPositiveDescription
End If

If cboNewWays.ListIndex = -1 Then
    tmpNewWays = ""
Else
    tmpNewWays = cboNewWays.ItemData(cboNewWays.ListIndex)
End If

If cboFrequency1.ListIndex = -1 Then
    tmpNewTechniques = ""
Else
    tmpNewTechniques = cboFrequency1.ItemData(cboFrequency1.ListIndex)
End If

If cboNoTechniques.ListIndex = -1 Then
    tmpNoTechniques = ""
Else
    tmpNoTechniques = cboNoTechniques.ItemData(cboNoTechniques.ListIndex)
End If

If cboCantDo.ListIndex = -1 Then
    tmpCantDo = ""
Else
    tmpCantDo = cboCantDo.ItemData(cboCantDo.ListIndex)
End If

For j = 0 To lstAreas(1).ListCount - 1
    If lstAreas(1).Selected(j) Then
        tmpBuf1 = tmpBuf1 & "1"
    Else
        tmpBuf1 = tmpBuf1 & "0"
    End If
Next

tmpAreas = tmpBuf1
tmpBuf1 = ""

If Trim(txtCDescription) = "" Then
    tmpCDescription = ""
Else
    tmpCDescription = txtCDescription
End If

If cboVisionChanged.ListIndex = -1 Then
    tmpVisionChanged = ""
Else
    tmpVisionChanged = cboVisionChanged.ItemData(cboVisionChanged.ListIndex)
End If

If cboHowChanged.ListIndex = -1 Then
    tmpHowChanged = ""
Else
```



```

    tmpHowChanged = cboHowChanged.ItemData(cboHowChanged.ListIndex)
End If

If cboMakeAppointment.ListIndex = -1 Then
    tmpMakeAppointment = ""
Else
    tmpMakeAppointment = cboMakeAppointment.ItemData(cboMakeAppointment.ListIndex)
End If

If cboActivity.ListIndex = -1 Then
    tmpPostActivity = ""
Else
    tmpPostActivity = cboActivity.ItemData(cboActivity.ListIndex)
End If

If cboFrequency2.ListIndex = -1 Then
    tmpPostSocial = ""
Else
    tmpPostSocial = cboFrequency2.ItemData(cboFrequency2.ListIndex)
End If

If cboEffect.ListIndex = -1 Then
    tmpPostFunctioning = ""
Else
    tmpPostFunctioning = cboEffect.ItemData(cboEffect.ListIndex)
End If

If Trim(txtEndDate2) = "" Then
    tmpEndDate2 = "12:00:00 PM"
Else
    tmpEndDate2 = txtEndDate2
End If

If (rsQA.BOF = False And rsQA.EOF = False) Then

    cnn.Execute "UPDATE QA Set ParticipantNumber = '" & tmpParticipantNumber & "'," & _
        " InterviewDate2 = '" & tmpInterviewDate2 & "', Interviewer2 = '" & tmpStaff & "'," & _
        " Positive = '" & tmpPositive & "', DPositive = '" & tmpDPositive & "'," & _
        " PositiveDescription = '" & tmpPositiveDescription & "', NewWays = '" & tmpNewWays & "'," & _
        " NewTechniques = '" & tmpNewTechniques & "', NoTechniques = '" & tmpNoTechniques & "'," & _
        " CantDo = '" & tmpCantDo & "', CantDoThings = '" & tmpAreas & "', CDescription = '" & tmpCD
        " ription & "'," & _
        " VisionChanged = '" & tmpVisionChanged & "', HowChanged = '" & tmpHowChanged & "'," & _
        " MakeAppointment = '" & tmpMakeAppointment & "', PostActivity = '" & tmpPostActivity & "'," & _
        " PostSocial = '" & tmpPostSocial & "', PostFunctioning = '" & tmpPostFunctioning & "'," & _
        " EndDate2 = '" & tmpEndDate2 & "' WHERE ConsumerID = '" & CustomerID & "' AND Agency = '" & A
        " cyCode & "'" & _
        " AND Episode = '" & Episode & "' AND CaseNumber = '" & CaseNumber & "'"

    MsgBox "Post Episode Survey successfully saved!", vbOKOnly, "Save"

    Exit Sub

End If

errSave:
    MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub

Private Sub Form_Load()

    Dim tmpBuf As String
    Dim i As Integer

    txtCaseNumber.Text = CaseNumber
    txtEpisode.Text = Episode
    txtConsumerName.Text = FirstName & " " & LastName

    txtPhoneNumber = PhoneNo

```

```
AgencyName = frmMainMenu.txtAgencyName
```

```
LoadControlsItems cboStaffs, False
LoadControlsItems cboFrequency1, False
LoadControlsItems cboNoTechniques, False
LoadControlsItems cboHowChanged, False
LoadControlsItems cboActivity, False
LoadControlsItems cboFrequency2, False
LoadControlsItems cboEffect, False
```

```
lblPositive.Caption = "Did the services you received from " & AgencyName & " make a positive difference in your ability to do things?"
```

```
lblNewWays.Caption = "When you received services from " & AgencyName & " were you shown new ways of doing things?"
```

```
lblFunctioning.Caption = "How much have " & AgencyName & " services affected your ability to function independently?"
```

```
Set rsQA = New Recordset
```

```
rsQA.Open "SELECT ParticipantNumber, InterviewDate2, Interviewer2," & _
         " Positive, DPositive, PositiveDescription, NewWays, NewTechniques," & _
         " NoTechniques, CantDo, CantDoThings, CDescription, VisionChanged, HowChanged" & _
         " MakeAppointment, PostActivity, PostSocial," & _
         " PostFunctioning, EndDate2" & _
         " FROM QA" & _
         " WHERE ConsumerID = " & CustomerID & " AND Agency = '" & AgencyCode & "'" & _
         " AND Episode = " & Episode & " AND CaseNumber = '" & CaseNumber & "'", cnn, adOpenStatic, adLockOptimistic
```

```
With rsQA
```

```
    If Not (.EOF And .BOF) Then
        .MoveFirst
```

```
    LoadDataToControls rsQA, txtParticipantNumber, False
```

```
    ' For textboxes we don't use LoadDataToControls
    ' because VB cannot accept nulls as values
    ' check for nulls or 12:00:00 PM in the database fields
    ' so we set the field to an empty string ("") or the field value
```

```
    If ![InterviewDate2] = "12:00:00 PM" Or Trim(![InterviewDate2] & vbNullString = "") Then
        txtInterviewDate2 = ""
    Else
        txtInterviewDate2 = ![InterviewDate2]
    End If
```

```
    If ![PositiveDescription] = "12:00:00 PM" Or Trim(![PositiveDescription] & vbNullString = "") Then
        Then
            txtPositiveDescription = ""
        Else
            txtPositiveDescription = ![PositiveDescription]
        End If
```

```
    If ![CDescription] = "12:00:00 PM" Or Trim(![CDescription] & vbNullString = "") Then
        txtCDescription = ""
    Else
        txtCDescription = ![CDescription]
    End If
```

```
    If ![EndDate2] = "12:00:00 PM" Or Trim(![EndDate2] & vbNullString = "") Then
        txtEndDate2 = ""
    Else
        txtEndDate2 = ![EndDate2]
        cmdSave.Enabled = False
    End If
```

```
    tmpBuf = ![Areas] & vbNullString
```

```
For i = 1 To Len(tmpBuf)
    If Mid(tmpBuf, i, 1) = "1" Then
        lstAreas1.Selected(i - 1) = True
    Else
        lstAreas1.Selected(i - 1) = False
    End If
Next

LoadDataToControls rsQA, cboStaffs, False
LoadDataToControls rsQA, cboPositive, False
LoadDataToControls rsQA, lstAreas, True
LoadDataToControls rsQA, cboFrequency1, False
LoadDataToControls rsQA, cboNoTechniques, False
LoadDataToControls rsQA, cboVisionChanged, False
LoadDataToControls rsQA, cboHowChanged, False
LoadDataToControls rsQA, cboMakeAppointment, False
LoadDataToControls rsQA, cboActivity, False
LoadDataToControls rsQA, cboFrequency2, False
LoadDataToControls rsQA, cboEffect, False
ElseIf (.BOF And .EOF) Then
    MsgBox "Please complete the End of Episode Survey first", vbOKOnly, "PostEpisode"
    Unload Me
End If

End With

End Sub

Private Sub Form_Unload(Cancel As Integer)
    rsQA.Close
End Sub
```

```
Option Explicit
Private InterventionCode(100) As String
Private IDescription(100) As String
Private rs As ADODB.Recordset
```

```
Private Sub cmdClose_Click()
    Unload Me
End Sub
```

```
Private Sub cmdSave_Click()
    SaveToDB
End Sub
```

```
Private Sub SaveToDB()
    Dim tmpIntervention As String
    Dim tmpStartDate As Date
    Dim tmpEndDate As Date
    Dim tmpStaff As String
    Dim tmpStr As String
    Dim CloseDate As Date
    Dim Itemindex As Integer
    Dim rs1 As ADODB.Recordset
```

```
    tmpEndDate = "12:00:00 PM"
```

```
    If cboInterventionCodes.ListIndex = -1 Then
        MsgBox "An Intervention is required to save this screen", vbOKOnly, "Missing Intervention"
        cboInterventionCodes.SetFocus
    Exit Sub
```

```
    Else
        Itemindex = cboInterventionCodes.ListIndex
        tmpIntervention = InterventionCode(Itemindex)
    End If
```

```
' Intervention selected in the combo box
' The intervention code in the array
' that matches what the user selected
```

```
    Set rs1 = New ADODB.Recordset
```

```
    tmpStr = "SELECT EndDate FROM Interventions WHERE ConsumerID = " & CustomerID & _
        " AND Agency = " & AgencyCode & " " & " AND Episode = " & txtEpisode & _
        " AND CaseNumber = " & txtCaseNumber & " AND InterventionID = " & tmpInterventio
    & " "
```

```
    rs1.Open tmpStr, cnn, adOpenForwardOnly, adLockReadOnly
```

```
    If Not rs1.EOF Then
        If (rs1![EndDate] = "12:00:00 PM") Then
            MsgBox IDescription(Itemindex) & " is already opened to the consumer", vbOKOnly, "Interv
ion already opened"
            cboInterventionCodes.SetFocus
            GoTo Finishup
        End If
    End If
```

```
    If Trim(txtStartDate) = "" Then
        MsgBox "A Start Date is required to save this screen", vbOKOnly, "Missing Start Date"
        txtStartDate.SetFocus
        GoTo Finishup
    Else
```

```
        tmpStartDate = txtStartDate.Text
    End If
```

```
    If cboStaffs.ListIndex = -1 Then
        MsgBox "A Staff Name is required to save this screen", vbOKOnly, "Missing Staff Name"
        cboStaffs.SetFocus
        GoTo Finishup
    Else
```

```
        tmpStaff = cboStaffs.ItemData(cboStaffs.ListIndex)
    End If
```

```
cnn.Execute "INSERT INTO Interventions " & _
    "(ConsumerId, Agency, Episode, CaseNumber, InterventionID, StartDate, EndDate, s
```

```
fName, EntryDate) VALUES (' &
    CustomerID & ', ' & AgencyCode & ', ' & txtEpisode & ', ' & txtCaseNumber & '
' & tmpIntervention & ', ' & tmpStartDate & ', ' & tmpEndDate & ', ' & tmpStaff & ', '
'Now()')
```

```
cmdSave.Enabled = False
cmdClose.SetFocus
MsgBox "Successfully Saved!", vbOKOnly, "Intervention Referral"
```

```
Finishup:
    rs1.Close
    Set rs1 = Nothing
    Exit Sub
End Sub
```

```
Private Sub Form_Load()
```

```
    Dim x As Integer
```

```
    x = 0
```

```
    ' Get the consumer name and episode from the Interventions form
    txtLastName = frmIntervention1.txtLastName
    txtFirstName = frmIntervention1.txtFirstName
    txtCaseNumber = frmIntervention1.txtCaseNumber
    txtEpisode = frmIntervention1.txtEpisode
```

```
    LoadControlsItems cboInterventionCodes, False
    LoadControlsItems cboStaffs, False
```

```
    Set rs = New ADODB.Recordset
    rs.Open "Select Code, Description From InterventionCodes", cnn, adOpenForwardOnly, adLockRea
ly
```

```
    rs.MoveFirst
    Do While Not rs.EOF
        InterventionCode(x) = rs![Code]
        IDescription(x) = rs![Description]
        rs.MoveNext
        x = x + 1
    Loop
```

```
    rs.Close
```

```
End Sub
```

```
Option Explicit
Private IsArrowKey As Integer
Private IsNewCustomer As Boolean
Private rstReg As ADODB.Recordset

Private Sub cboBiggestProblem_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboBiggestProblem.ListIndex = -1
End If
End Sub

Private Sub cboBiggestProblem_LostFocus()

If Not CheckForShiftTab Then
    If cboBiggestProblem.ListIndex <> 7 Then
        cboYesno(0).SetFocus
    End If
End If
End Sub

Private Sub cboCallerReason_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboCallerReason.ListIndex = -1
End If
End Sub

Private Sub cboCaseStatus_Click()

If IsArrowKey > 0 Then Exit Sub

If cboCaseStatus.ListIndex <> 2 Then
    cboCallerReason.ListIndex = -1
    cboStaffs.SetFocus
End If
End Sub

Private Sub cboCaseStatus_KeyDown(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub

Private Sub cboCaseStatus_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboCaseStatus.ListIndex = -1
End If
End Sub

Private Sub cboCaseStatus_LostFocus()

If CheckForTab Then
    If cboCaseStatus.ListIndex <> 2 Then cboStaffs.SetFocus
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboCorrespondence_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
```

```
cboCorrespondence.ListIndex = -1
End If
End Sub

Private Sub cboLanguages_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboLanguages.ListIndex = -1
End If
End Sub

Private Sub cboReferralSource_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboReferralSource.ListIndex = -1
End If
End Sub

Private Sub cboRelationship_Click()

If cboRelationship.ListIndex <> 0 Then
    txtRegistration(8) = ""
    txtRegistration(9) = ""
    txtRegistration(10) = ""
    txtRegistration(11) = ""
End If
End Sub

Private Sub cboRelationship_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboRelationship.ListIndex = -1
End If
End Sub

Private Sub cboRelationship_LostFocus()

If Not CheckForShiftTab Then
    If cboRelationship.ListIndex = 0 Then
        txtRegistration(8) = txtRegistration(3)
        txtRegistration(9) = txtRegistration(4)
        txtRegistration(10) = txtRegistration(5)
        txtRegistration(11) = txtRegistration(6)
    End If
End If
End Sub

Private Sub cboStaffs_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboStaffs.ListIndex = -1
End If
End Sub

Private Sub cboYesNo_Click(Index As Integer)

If IsArrowKey > 0 Then Exit Sub

If Index = 0 Then
    If cboyesno(Index).ListIndex = 0 Then
        lstCrises.SetFocus
    Else
        lstUrgency.SetFocus
    End If
ElseIf Index = 1 Then
    If cboyesno(Index).ListIndex = 0 Then
        txtRegistration(13) = txtRegistration(12)
        txtRegistration(21) = txtRegistration(19)
        txtRegistration(0) = txtRegistration(20)
        txtRegistration(23) = txtRegistration(22)
    End If
End If
End Sub
```

```
Else
    txtRegistration(14).SetFocus
End If
ElseIf Index = 3 Then
    If cboYesno(Index).ListIndex = 0 Then
        txtRegistration(7) = ""
        cboRelationship.ListIndex = 0
        cboRelationship.SetFocus
    End If
End If
End Sub

Private Sub cboYesNo_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub

Private Sub cboYesNo_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboYesno(Index).ListIndex = -1
End If
End Sub

Private Sub cboYesNo_LostFocus(Index As Integer)

If Index = 0 Then
    If CheckForTab Then
        If cboYesno(Index).ListIndex = 0 Then
            lstCrises.SetFocus
        Else
            lstUrgency.SetFocus
        End If
    End If
ElseIf Index = 1 Then
    If CheckForTab Then
        If cboYesno(Index).ListIndex = 0 Then
            txtRegistration(13) = txtRegistration(12)
            txtRegistration(21) = txtRegistration(19)
            txtRegistration(0) = txtRegistration(20)
            txtRegistration(23) = txtRegistration(22)
        Else
            txtRegistration(14).SetFocus
        End If
    End If
ElseIf Index = 3 Then
    If CheckForTab Then
        If cboYesno(Index).ListIndex = 0 Then
            cboRelationship.ListIndex = 0
            cboRelationship.SetFocus
        End If
    End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboYesno_Validate(Index As Integer, Cancel As Boolean)

If Index = 0 Then
    If cboYesno(Index).ListIndex = -1 Then
        MsgBox "Please had to select one of them!", vbInformation + vbOKOnly, "In Crisis"
    End If
End If
End Sub
```



```

        Cancel = True
    End If
End If
End Sub

Private Sub cmdClose_Click()
    'frmMainMenu.Show
    Unload Me
End Sub

Private Sub cmdInstructpg1_Click()
    Dim response As Integer

    tabRegistration.Tab = 0
    response = MsgBox("Actions: If in crisis a Supervisor will immediately intervene. " & _
        "If services rejected, a Contact Person will call within 24 hours to" & _
        " follow-up and offer assistance. If urgent, call within 48 hours.", vbOKOnly
    "Crisis")
    If response = vbOK Then tabRegistration.Tab = 1

End Sub

Private Sub cmdInstructpg1_LostFocus()

If CheckForShiftTab Then
    tabRegistration.Tab = 0
    lstUrgency.SetFocus
End If
End Sub

Private Sub cmdNextPage_Click()

tabRegistration.Tab = 1
End Sub

Private Sub cmdPrevious_Click()
tabRegistration.Tab = 0
End Sub

Private Sub cmdSave_Click()
    Dim colReg As New Collection
    Dim i As Integer, j As Integer
    Dim tmpStr As String
    'Dim rst As ADODB.Recordset
    On Error GoTo errSave

    If Trim(txtRegistration(8)) = "" Or Trim(txtRegistration(10)) = "" Then
        MsgBox "Please enter both first name and last name", vbInformation + vbOKOnly, "Registration"
        Exit Sub
    ElseIf IsNewCustomer Then
        'Set rst = New ADODB.Recordset
        'rst.Open "Select ConsumerID, FinishDate From Registration Where VIFirstName = '" & Trim(txtRegistration(8)) & _
        " '" And VILastName = '" & Trim(txtRegistration(10)) & '"", cnn, adOpenForwardOnly, adLockReadOnly
        'If Not (rst.BOF And rst.EOF) Then
        'MsgBox "The name '" & txtRegistration(8) & "' & txtRegistration(10) & "' has registered already!", _
        vbInformation, "registration"
        'tabRegistration.Tab = 0
        'Exit Sub
        'End If
        'Set rst = Nothing
    End If

    colReg.Add cboYesno
    colReg.Add txtRegistration
    colReg.Add cboRelationship
    colReg.Add lstCrises
    colReg.Add lstUrgency

```

```
colReg.Add cboBiggestProblem
colReg.Add cboReferralSource
colReg.Add cboCorrespondence
colReg.Add cboLanguages
colReg.Add cboCaseStatus
colReg.Add cboCallerReason
colReg.Add cboStaffs
colReg.Add txtAgency
```

```
With rstReg
  If (.EOF And .BOF) Then .AddNew
  For i = 1 To colReg.Count
    If i <= 2 Then 'for control collection case
      If TypeOf colReg(i) Is TextBox Then
        For j = 0 To colReg(i).Count - 1
          If colReg(i)(j).Tag = "FinishDate" Then
            If Trim(colReg(i)(j)) = "" Then
              .Fields.Item(colReg(i)(j).Tag) = "12:00:00 PM"
            Else
              .Fields.Item(colReg(i)(j).Tag) = CDate(colReg(i)(j))
            End If
          Else
            .Fields.Item(colReg(i)(j).Tag) = Trim(colReg(i)(j))
          End If
        Next
      Else
        For j = 0 To colReg(i).Count - 1
          If colReg(i)(j).ListIndex = -1 Then
            .Fields.Item(colReg(i)(j).Tag) = ""
          Else
            .Fields.Item(colReg(i)(j).Tag) = colReg(i)(j).ItemData(colReg(i)(j).ListIndex)
          End If
        Next
      End If
    Else
      If TypeOf colReg(i) Is ListBox Then
        For j = 0 To colReg(i).ListCount - 1
          If colReg(i).Selected(j) Then
            tmpStr = tmpStr & "1"
          Else
            tmpStr = tmpStr & "0"
          End If
        Next
        .Fields.Item(colReg(i).Tag) = tmpStr
        tmpStr = ""
      ElseIf TypeOf colReg(i) Is TextBox Then
        .Fields.Item(colReg(i).Tag) = Trim(colReg(i))
      Else
        If colReg(i).ListIndex = -1 Then
          .Fields.Item(colReg(i).Tag) = ""
        Else
          .Fields.Item(colReg(i).Tag) = colReg(i).ItemData(colReg(i).ListIndex)
        End If
      End If
    End If
  Next
  .Update
End With

Me.Refresh
MsgBox "Successfully saved the information!", vbOKOnly, "Save"
Set colReg = Nothing
Exit Sub

errSave:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub

Private Sub Form_Activate()
```

On Error GoTo Errhandler

txtAgency = AgencyCode

'lblReferralSource.Caption = "p. How did you learn about the " & frmSearchMenu.txtAgencyName.Text & "?"

'If frmSearch.txtTotalConsumers = "0" Then

'txtRegistration(1) = frmSearch.txtCaseNumber

'txtRegistration(8) = frmSearch.txtFirstName

'txtRegistration(10) = frmSearch.txtLastName

'End If

LoadControlsItems cboStaffs, False

Set rstReg = New ADODB.Recordset

rstReg.CursorLocation = adUseClient

rstReg.Open "Select \* from Registration Where ConsumerID=" & CustomerID, cnn, adOpenDynamic, ad  
ckOptimistic

With rstReg

If Not (.EOF And .BOF) Then

'Have record for this client and load it to f

m

.MoveFirst

LoadDataToControls rstReg, cboYesno, True

LoadDataToControls rstReg, lstCrises, False

LoadDataToControls rstReg, lstUrgency, False

LoadDataToControls rstReg, cboRelationship, False

LoadDataToControls rstReg, txtRegistration, True

LoadDataToControls rstReg, txtAgency, False

LoadDataToControls rstReg, cboBiggestProblem, False

LoadDataToControls rstReg, cboReferralSource, False

LoadDataToControls rstReg, cboCorrespondence, False

LoadDataToControls rstReg, cboLanguages, False

LoadDataToControls rstReg, cboCaseStatus, False

LoadDataToControls rstReg, cboCallerReason, False

LoadDataToControls rstReg, cboStaffs, False

Else

IsNewCustomer = True

End If

End With

tabRegistration.Tab = 0

txtAgency.SetFocus

Exit Sub

Errhandler:

MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"

End Sub

Private Sub Form\_Unload(Cancel As Integer)

'If rstReg.State = adStateOpen Then rstReg.Close

FirstName = Trim(txtRegistration(8))

LastName = Trim(txtRegistration(10))

CaseNumber = Trim(txtRegistration(1))

PhoneNo = Trim(txtRegistration(11))

Address = Trim(txtRegistration(12)) & ", " & Trim(txtRegistration(19)) & ", " & Trim(txtRegistration(22)) & " " & Trim(txtRegistration(20))

If Trim(Address) = " , " Then Address = ""

Set rstReg = Nothing

Set frmRegistration1 = Nothing

End Sub

Private Sub lstCrises\_KeyUp(KeyCode As Integer, Shift As Integer)

Dim i As Integer

If KeyCode = vbKeyBack Then

If lstCrises.SelCount = 1 Then

lstCrises.Selected(lstCrises.ListIndex) = False

ElseIf lstCrises.ListCount > 1 Then

```

    For i = lstCrises.ListCount - 1 To 0 Step -1
        If lstCrises.Selected(i) Then
            lstCrises.Selected(i) = False
        End If
    Next
End If
End If
End Sub

Private Sub lstUrgency_KeyUp(KeyCode As Integer, Shift As Integer)
    Dim i As Integer

    If KeyCode = vbKeyBack Then
        If lstUrgency.SelCount = 1 Then
            lstUrgency.Selected(lstUrgency.ListIndex) = False
        ElseIf lstUrgency.ListCount > 1 Then
            For i = lstUrgency.ListCount - 1 To 0 Step -1
                If lstUrgency.Selected(i) Then
                    lstUrgency.Selected(i) = False
                End If
            Next
        End If
    End If
End If
End Sub

Private Sub lstUrgency_LostFocus()

    If CheckForTab Then
        tabRegistration.Tab = 1
    End If
End Sub

Private Sub tabRegistration_Click(PreviousTab As Integer)

    If tabRegistration.Tab = 1 Then
        cmdInstructpg1.SetFocus
        frmRegistration1.Caption = "Registration - Basic Information 2"
        cmdNextPage.Enabled = False
        cmdPrevious.Enabled = True
    ElseIf tabRegistration.Tab = 0 Then
        'txtAgency.SetFocus
        frmRegistration1.Caption = "Registration - Basic Information 1"
        cmdPrevious.Enabled = False
        cmdNextPage.Enabled = True
    End If
End Sub

Private Sub txtRegistration_Change(Index As Integer)

    If Index = 17 Then
        If Len(txtRegistration(Index)) = 3 Then
            If Not IsNumeric(Mid(txtRegistration(Index), 1, 3)) Then
                MsgBox "Please enter numeric number for Social Security number!", vbInformation + vbO
nly, " Phone Number"
                txtRegistration(Index) = ""
                txtRegistration(Index).SetFocus
            Else
                txtRegistration(Index) = txtRegistration(Index) & "--"
                txtRegistration(Index).SelStart = 5
            End If
        ElseIf Len(txtRegistration(Index)) = 6 Then
            If Not IsNumeric(Mid(txtRegistration(Index), 5, 2)) Then
                MsgBox "Please enter numeric number for Social Security number!", vbInformation + vbO
nly, " Phone Number"
                txtRegistration(Index) = ""
                txtRegistration(Index).SetFocus
            Else
                txtRegistration(Index) = txtRegistration(Index) & "--"
                txtRegistration(Index).SelStart = 8
            End If
        End If
    End If
End Sub

```

```

ElseIf Len(txtRegistration(Index)) = 11 Then
    If Not IsNumeric(Mid(txtRegistration(Index), 8, 4)) Then
        MsgBox "Please enter numeric number for Social Security number!", vbInformation + vbOnly, " Phone Number"
        txtRegistration(Index) = ""
        txtRegistration(Index).SetFocus
    End If
End If
ElseIf Index = 6 Or Index = 11 Or Index = 14 Or Index = 15 Then
    If Len(txtRegistration(Index)) = 4 Then
        If Not IsNumeric(Mid(txtRegistration(Index), 2, 3)) Then
            MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, " Phone Number"
            txtRegistration(Index) = Mid(txtRegistration(Index), 1, 2)
            txtRegistration(Index).SelStart = 2
        Else
            txtRegistration(Index) = txtRegistration(Index) & "-"
            txtRegistration(Index).SelStart = 7
        End If
    ElseIf Len(txtRegistration(Index)) = 9 Then
        If Not IsNumeric(Mid(txtRegistration(Index), 7, 3)) Then
            MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, " Phone Number"
            txtRegistration(Index) = Mid(txtRegistration(Index), 1, 6)
            txtRegistration(Index).SelStart = 7
        Else
            txtRegistration(Index) = txtRegistration(Index) & "-"
            txtRegistration(Index).SelStart = 11
        End If
    ElseIf Len(txtRegistration(Index)) = 14 Then
        If Not IsNumeric(Mid(txtRegistration(Index), 11, 3)) Then
            MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, " Phone Number"
            txtRegistration(Index) = Mid(txtRegistration(Index), 1, 10)
            txtRegistration(Index).SetFocus
        End If
    End If
End If
End Sub

```

```
Private Sub txtRegistration_GotFocus(Index As Integer)
```

```

If Index = 6 Or Index = 11 Or Index = 14 Or Index = 15 Then
    If txtRegistration(Index) = "" Then
        txtRegistration(Index) = "("
        txtRegistration(Index).SelStart = 2
    End If
End If
End Sub

```

```
Private Sub txtRegistration_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```

If Index = 6 Or Index = 11 Or Index = 14 Or Index = 15 Then
    If KeyCode = vbKeyBack Then
        txtRegistration(Index) = "("
        txtRegistration(Index).SelStart = 2
    End If
ElseIf Index = 17 Then
    If KeyCode = vbKeyBack Then txtRegistration(Index) = ""
End If
End Sub

```

```
Private Sub txtRegistration_LostFocus(Index As Integer)
```

```

If Index = 6 Or Index = 11 Or Index = 14 Or Index = 15 Then
    If txtRegistration(Index) = "(" Then txtRegistration(Index) = ""
End If
End Sub

```

```
Private Sub txtRegistration_Validate(Index As Integer, Cancel As Boolean)
```

```
Dim rst As ADODB.Recordset
```

```
If Index = 1 Then
```

```
    If txtRegistration(Index) <> "" And IsNewCustomer Then
```

```
        Set rst = New ADODB.Recordset
```

```
        rst.Open "Select ConsumerID From Registration Where CaseNumber = '" & _  
            Trim(txtRegistration(Index)) & "'", cnn, adOpenForwardOnly, adLockReadOnly.
```

```
        If Not (rst.EOF And rst.BOF) Then
```

```
            MsgBox "The CaseNumber " & Trim(txtRegistration(Index)) & " has been used, please select  
another one", _
```

```
                vbInformation + vbOKOnly, "Case Number"
```

```
            txtRegistration(Index) = ""
```

```
            Cancel = True
```

```
        End If
```

```
        Set rst = Nothing
```

```
    End If
```

```
ElseIf Index = 2 Then
```

```
    If txtRegistration(Index) <> "" Then
```

```
        If Not IsDate(txtRegistration(Index)) Then
```

```
            MsgBox "Please enter date in this field", vbInformation + vbOKOnly, "Registration Date"
```

```
            Cancel = True
```

```
        End If
```

```
    End If
```

```
ElseIf Index = 20 Or Index = 0 Then
```

```
    If txtRegistration(Index) <> "" Then
```

```
        If Not IsNumeric(txtRegistration(Index)) Then
```

```
            MsgBox "Please enter numeric number in this field", vbInformation + vbOKOnly, "Registr
```

```
ion Date"
```

```
            Cancel = True
```

```
        End If
```

```
    End If
```

```
End If
```

```
End Sub
```

```
Option Explicit
Private rstReport As ADODB.Recordset
Private rstTemp As ADODB.Recordset
```

```
Private Sub cmdActivityReport_Click()
```

```
If Not IsDate(Trim(txtPeriodFrom)) Then
    MsgBox "Please enter date for Reporting Period!", vbInformation, "Reporting Period"
    txtPeriodFrom = ""
    txtPeriodFrom.SetFocus
Exit Sub
```

```
ElseIf Not IsDate(Trim(txtPeriodTo)) Then
    MsgBox "Please enter date for Reporting Period!", vbInformation, "Reporting Peri"
    txtPeriodTo = ""
    txtPeriodTo.SetFocus
Exit Sub
```

```
End If
```

```
If Printers.Count = 0 Then
    MsgBox "Please add one printer to your computer!", vbOKOnly + vbInformation, "Add Printer"
Exit Sub
End If
```

```
cnn.Execute "Delete * from AgencyActivityReport"
Set rstReport = cnn.OpenSchema(adSchemaTables, Array(Empty, Empty, "tmpTable", "Table"))
If Not rstReport.EOF Then cnn.Execute "Drop table tmpTable"
```

```
'For all Interventions
```

```
If rstReport.State = adStateOpen Then rstReport.Close
cnn.Execute "Select Distinct consumerID Into tmpTable From Visits Where ( ServiceDate >= #" & _
    Trim(txtPeriodFrom) & "# And ServiceDate <= #" & txtPeriodTo & "#) And (Duration Is Not Null" & _
    "nd duration <> '')"
```

```
cnn.Execute "Insert Into AgencyActivityReport(AllConsumers) Select Count(ConsumerId) From tmpTa" & _
    "e"
```

```
rstReport.Open "Select * from AgencyActivityReport", cnn, adOpenDynamic, adLockOptimistic
rstTemp.Open "Select Sum(IIF((isdate(EpisodeCloseddate) And EpisodeCloseddate <> #12:00:00 PM#), 0) AS EpisodeClosed, " & _
    "Sum(IIF(EpisodeOutcome = '12', 1,0) ) As EpisodeOutcome1 FROM Triage Where (EpisodeCloseddat" & _
    ">= #" & Trim(txtPeriodFrom) & "# And EpisodeCloseddate <= #" & Trim(txtPeriodTo) & "#)", cnn, adOpenForwardOnly, adLockRead" & _
    "ly
```

```
rstReport![PeriodFrom] = Trim(txtPeriodFrom)
rstReport![PeriodTo] = Trim(txtPeriodTo)
rstReport![EpisodeClosed] = IIf(IsNull(rstTemp![EpisodeClosed]), 0, 1)
rstReport![EpisodeOutcome] = IIf(IsNull(rstTemp![EpisodeOutcome1]), 0, 1)
rstReport![SucessRate] = IIf(rstTemp![EpisodeClosed] = "0", "0", Format((rstTemp![EpisodeOutcc" & _
    "1] / rstTemp![EpisodeClosed]), "0.0%"))
If rstReport![EpisodeClosed] = "0" Then
    rstReport![SucessRate] = "0"
Else
    rstReport![SucessRate] = Format((rstReport![EpisodeOutcome] / rstReport![EpisodeClosed]), "0" & _
    "%")
```

```
End If
```

```
rstReport.Update
```

```
rstTemp.Close
```

```
cnn.Execute "Drop Table tmpTable"
```

```
Call InterventionActivity("SocialConsumers", "SocialClosed", "SocialOutcome", "SocialSucessRate" & _
    "010")
```

```
'Call InterventionActivity("PSConsumers", "PSClosed", "PSOutcome", "PSSucessRate", "011")
```

```
Call InterventionActivity("ILConsumers", "ILClosed", "ILOutcome", "ILSucessRate", "013")
```

```
Call InterventionActivity("OMConsumers", "OMClosed", "OMOutcome", "OMSucessRate", "014")
```

```
'Call InterventionActivity("CSConsumers", "CSClosed", "CSOutcome", "CSSucessRate", "031")
```

```
Call InterventionActivity("LVCConsumers", "LVClosed", "LVOutcome", "LVSucessRate", "041")
```

```
'Call InterventionActivity("IDConsumers", "IDClosed", "IDOutcome", "IDSucessRate", "045")
```

```
'Call InterventionActivity("DMConsumers", "DMClosed", "DMOutcome", "DMSucessRate", "047")
```

```
Call InterventionActivity("EmploymentConsumers", "EmploymentClosed", "EmploymentOutcome", "Emp" & _
    "mentSucessRate", "050")
```

```
'Call InterventionActivity("ASConsumers", "ASClosed", "ASOutcome", "ASSucessRate", "056")
```

```
'Call InterventionActivity("MUConsumers", "MUClosed", "MUOutcome", "MUSucessRate", "121")
```

```
'Call InterventionActivity("RSConsumers", "RSClosed", "RSOutcome", "RSSuccessRate", "141")
Call InterventionActivity("ComputerConsumers", "ComputerClosed", "ComputerOutcome", "ComputerSuccessRate", "160")
```

```
Set rptAgencyActivity.DataSource = rstReport
rptAgencyActivity.PrintReport
End Sub
```

```
Private Sub cmdClose_Click()
Unload Me
End Sub
```

```
Private Sub InterventionActivity(strConsumer As String, strClosed As String, strOutcome As String, strSuccessRate As String, tmpID As String)
```

```
cnn.Execute "Select Distinct consumerID Into tmpTable From Visits Where ( ServiceDate >= #" & _
Trim(txtPeriodFrom) & "# And ServiceDate <= #" & txtPeriodTo & "#) And InterventionID = '" & _
tmpID & "'"
rstTemp.Open "Select Count(ConsumerId) From tmpTable", cnn, adOpenForwardOnly, adLockReadOnly
rstReport.Fields(strConsumer) = rstTemp.Fields(0)
rstReport.Update
rstTemp.Close
rstTemp.Open "Select Sum(IIF((isdate(EndDate) And EndDate <> #12:00:00 PM#), 1, 0)) AS Closed,
& " Sum(IIF(InterventionOutcome = '12', 1,0) ) AS OutCome FROM Interventions Where ( EndDate >
#" & _
Trim(txtPeriodFrom) & "# And EndDate <= #" & txtPeriodTo & "#) And InterventionID = '" & tmp
& "'", cnn, adOpenForwardOnly, adLockReadOnly
rstReport.Fields(strClosed) = IIf(IsNull(rstTemp![Closed]), 0, 1)
rstReport.Fields(strOutcome) = IIf(IsNull(rstTemp![Outcome]), 0, 1)
If rstReport.Fields(strClosed) = "0" Then
rstReport.Fields(strSuccessRate) = "0.0%"
Else
rstReport.Fields(strSuccessRate) = Format((rstReport.Fields(strOutcome) / rstReport.Fields(strClosed)), "0.0%")
End If
rstReport.Update
rstTemp.Close
cnn.Execute "Drop Table tmpTable"
End Sub
```

```
Private Sub Form_Load()
```

```
Set rstTemp = New ADODB.Recordset
rstTemp.CursorLocation = adUseClient
```

```
Set rstReport = New ADODB.Recordset
rstReport.CursorLocation = adUseClient
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
If rstReport.State = adStateOpen Then rstReport.Close
Set rstReport = Nothing
Set rstTemp = Nothing
Set frmReportMenu = Nothing
End Sub
```



```
Option Explicit
Private rstSearch As ADODB.Recordset
```

```
Private Sub cmdAdd_Click()
    CustomerID = 0 'for existing record, the CustomerID had to be larger than 1
    frmRegistration1.Show
End Sub
```

```
Private Sub cmdClose_Click()
    Unload Me
End Sub
```

```
Private Sub cmdMainMenu_Click()
    frmMainMenu.Show 'vbModal
End Sub
```

```
Private Sub cmdNext_Click()
```

```
With rstSearch
    .MoveNext
    If .AbsolutePosition = .RecordCount Then cmdNext.Enabled = False
    txtCounter = .AbsolutePosition
    txtCaseNumber = ![CaseNumber]
    txtLastName = ![VILastName]
    txtFirstName = ![VIFirstName]
    CaseNumber = ![CaseNumber]
    CustomerID = ![consumerid]
    AgencyCode = ![Agency]
    FirstName = ![VIFirstName]
    LastName = ![VILastName]
    PhoneNo = ![VIPhone]
    Address = ![address1] & ", " & ![City1] & ", " & ![State1] & " " & ![Zip1]
    If Trim(Address) = ",," Then Address = ""
    cmdPrevious.Enabled = True
    txtFirstName.SetFocus
End With
End Sub
```

```
Private Sub cmdPrevious_Click()
```

```
With rstSearch
    .MovePrevious
    If .AbsolutePosition = 1 Then cmdPrevious.Enabled = False
    txtCounter = .AbsolutePosition
    txtCaseNumber = ![CaseNumber]
    txtLastName = ![VILastName]
    txtFirstName = ![VIFirstName]
    CaseNumber = ![CaseNumber]
    CustomerID = ![consumerid]
    AgencyCode = ![Agency]
    FirstName = ![VIFirstName]
    LastName = ![VILastName]
    PhoneNo = ![VIPhone]
    Address = ![address1] & ", " & ![City1] & ", " & ![State1] & " " & ![Zip1]
    If Trim(Address) = ",," Then Address = ""
    cmdNext.Enabled = True
    txtFirstName.SetFocus
End With
End Sub
```

```
Private Sub Form_Activate()
    Dim tmpStr As String
```

```
If rstSearch.State = adStateOpen Then rstSearch.Close
rstSearch.Open SQLString, cnn, adOpenForwardOnly, adLockReadOnly
Set SearchGrid.DataSource = rstSearch
```

```
With rstSearch
    If Not (.BOF And .EOF) Then
        .MoveFirst
    End If
End With
```

```
txtTotalConsumers = .RecordCount
txtCounter = .AbsolutePosition
txtCaseNumber = ![CaseNumber]
txtLastName = ![VILastName]
txtFirstName = ![VIFirstName]
CaseNumber = ![CaseNumber]
CustomerID = ![consumerid]
AgencyCode = ![Agency]
FirstName = ![VIFirstName]
LastName = ![VILastName]
PhoneNo = ![VIPhone]
Address = ![address1] & ", " & ![City1] & ", " & ![State1] & " " & ![Zip1]
txtCaseNumber = CaseNumber
txtLastName = LastName
txtFirstName = FirstName
If Trim(Address) = " , " Then Address = ""
If .RecordCount > 1 Then cmdNext.Enabled = True
cmdMainMenu.Enabled = True
Else
    txtTotalConsumers = 0
    txtCounter = 0
    txtCaseNumber = ""
    txtLastName = ""
    txtFirstName = ""
    cmdMainMenu.Enabled = False
    cmdNext.Enabled = False
    cmdPrevious.Enabled = False
End If
End With
txtTotalConsumers.SetFocus
End Sub

Private Sub Form_Load()

Set rstSearch = New ADODB.Recordset
rstSearch.CursorLocation = adUseClient

End Sub

Private Sub Form_Unload(Cancel As Integer)
If Forms.Count = 1 Then
    If cnn.State = adStateOpen Then cnn.Close
    Set cnn = Nothing
End If

If rstSearch.State = adStateOpen Then rstSearch.Close
Set rstSearch = Nothing
Set frmSearch = Nothing
End Sub

Private Sub SearchGrid_SelChange(Cancel As Integer)

If SearchGrid.SelBookmarks.Count = 0 Then Exit Sub
rstSearch.AbsolutePosition = SearchGrid.SelBookmarks(0)

With rstSearch
    If Not (.BOF And .EOF) Then
        txtCounter = .AbsolutePosition
        txtCaseNumber = ![CaseNumber]
        txtLastName = ![VILastName]
        txtFirstName = ![VIFirstName]
        CaseNumber = ![CaseNumber]
        CustomerID = ![consumerid]
        AgencyCode = ![Agency]
        FirstName = ![VIFirstName]
        LastName = ![VILastName]
        Address = ![address1] & ", " & ![City1] & ", " & ![State1] & " " & ![Zip1]
        If Trim(Address) = " , " Then Address = ""

        If .RecordCount > 1 Then
```

```
    If .AbsolutePosition = 1 Then
        cmdPrevious.Enabled = False
        cmdNext.Enabled = True
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPrevious.Enabled = True
        cmdNext.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPrevious.Enabled = True
        cmdNext.Enabled = False
    End If
End If

txtCounter.SetFocus

End If
End With
End Sub
```

Option Explicit

Private Sub cboInterventioncodes\_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then  
 cboInterventionCodes.ListIndex = -1  
End If  
End Sub

Private Sub cboStaffs\_KeyUp(KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then  
 cboStaffs.ListIndex = -1  
End If  
End Sub

Private Sub cmdClear\_Click()

cboInterventionCodes.ListIndex = -1  
 cboStaffs.ListIndex = -1  
 txtLastName.Text = ""  
 txtFirstName.Text = ""  
 txtCaseNumber.Text = ""

End Sub

Private Sub cmdClose\_Click()

Unload Me  
End Sub

Private Sub cmdReport\_Click()

frmReportMenu.Show vbModal  
End Sub

Private Sub cmdSearch\_Click()

LastName = Trim(txtLastName)  
 FirstName = Trim(txtFirstName)  
 CaseNumber = Trim(txtCaseNumber)

txtLastName = Replace(txtLastName, " ", " ")

If cboInterventionCodes.ListIndex <> -1 Then ' Select by Intervention and Staff (if exists)  
 SQLString = "SELECT DISTINCT R.VILastName, R.VIFirstName, R.CaseNumber, R.ConsumerID, R.VIPh  
e," & \_  
" I.InterventionID, R.Address1, R.City1, R.Statel, R.Zip1, R.Agency FROM Registra  
tion R, Interventions I" & \_  
" WHERE R.ConsumerID = I.ConsumerID AND I.InterventionID = (Select Code from Inte  
ventionCodes" & \_  
" Where Description = '" & cboInterventionCodes.Text & "') AND (I.EndDate Is Null  
OR I.EndDate = #12:00:00 PM#)" & \_  
" AND R.Agency = '" & txtAgencyCode & "'"

If cboStaffs.ListIndex <> -1 Then

SQLString = SQLString & " And I.StaffName = '" & cboStaffs.ItemData(cboStaffs.ListIndex) & "

End If

End If

If cboInterventionCodes.ListIndex = -1 And cboStaffs.ListIndex <> -1 Then ' Select by staff, no  
intervention

SQLString = "SELECT DISTINCT R.VILastName, R.VIFirstName, R.CaseNumber, R.ConsumerID, R.VIPh  
e," & \_  
" I.InterventionID, R.Address1, R.City1, R.Statel, R.Zip1, R.Agency, I.StaffName  
FROM Registration as R, Interventions I" & \_  
" WHERE R.ConsumerId = I.ConsumerID AND R.Agency = '" & txtAgencyCode & "' & \_  
" AND (I.EndDate Is Null OR I.EndDate = #12:00:00 PM#) AND I.StaffName = '" & \_  
cboStaffs.ItemData(cboStaffs.ListIndex) & "'"

End If

If cboStaffs.ListIndex = -1 And cboInterventionCodes.ListIndex = -1 Then ' No staff, no interve  
tion

SQLString = "SELECT DISTINCT R.VILastName, R.VIFirstName, R.CaseNumber, R.ConsumerID, R.Addre

```
sl, " & _
    " R.City1, R.State1, R.Zip1, R.VIPhone, R.Agency FROM Registration R WHERE R.Age
y = " & txtAgencyCode & ""
End If

If Trim(txtLastName.Text) <> "" Then
    SQLString = SQLString & " AND R.VILastName Like " & Trim(txtLastName) & "%'"
End If

If Trim(txtFirstName.Text) <> "" Then
    SQLString = SQLString & " AND R.VIFirstName Like '" & Trim(txtFirstName) & "%'"
End If

If Trim(txtCaseNumber.Text) <> "" Then
    SQLString = SQLString & " AND R.CaseNumber = " & Trim(txtCaseNumber) & ""
End If

SQLString = SQLString & " Order By R.VILastName"
frmSearch.Show vbModal
End Sub

Private Sub Form_Activate()
    cboInterventionCodes.ListIndex = -1
    cboStaffs.ListIndex = -1
    txtLastName.Text = ""
    txtFirstName.Text = ""
    txtCaseNumber.Text = ""
    txtFirstName.SetFocus
End Sub

Private Sub Form_Load()

    LoadControlsItems cboStaffs, False
    LoadControlsItems cboInterventionCodes, False

    AgencyCode = txtAgencyCode
End Sub

Private Sub Form_Unload(Cancel As Integer)
    If Forms.Count = 1 Then
        If cnn.State = adStateOpen Then cnn.Close
        Set cnn = Nothing
    End If
    Set frmSearchMenu = Nothing
End Sub
```

```
Option Explicit
Private IsArrowKey As Integer
Private IsHousingIssues As Boolean
Private rst As ADODB.Recordset
Private GoalPosition As Integer
Private TriggerPosition As Integer
Private rstSocial As ADODB.Recordset
Public IsNewRecord As Boolean
Private IsGoalsClicked As Boolean
Public IsFileDateChecked As Boolean
Private IsVisualInfoClicked As Boolean

Private Sub cboAgree_Click(Index As Integer)
Dim i As Integer

If IsArrowKey > 0 Then Exit Sub

Call cboAgree_Validate(Index, False)
If Index = 3 Or Index = 4 Or Index = 9 Or Index = 11 Or Index = 15 Or Index = 17 Or Index = 20
Or Index = 22 Or Index = 25 Or Index = 27 Then
For i = GoalPosition + 1 To chkRecommendation.Count - 1
If chkRecommendation(i).Value = 1 Then
GoalPosition = i
If i > 4 Then tabSocial.Tab = 8
chkRecommendation(i).SetFocus
Exit Sub
End If
Next

tabSocial.Tab = 8
Call CountScore
Call CountGoals
txtsocial(11).SetFocus
End If
End Sub

Private Sub cboAgree_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then
IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
IsArrowKey = KeyCode
End If
End Sub

Private Sub cboAgree_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
cboAgree(Index).ListIndex = 1
End If
End Sub

Private Sub cboAgree_LostFocus(Index As Integer)
Dim i As Integer

If CheckForTab Then
Call cboAgree_Validate(Index, False)
If Index = 3 Or Index = 4 Or Index = 9 Or Index = 11 Or Index = 15 Or Index = 17 Or Index =
Or Index = 22 Or Index = 25 Or Index = 27 Then
For i = GoalPosition + 1 To chkRecommendation.Count - 1
If chkRecommendation(i).Value = 1 Then
GoalPosition = i
If i > 4 Then tabSocial.Tab = 8
chkRecommendation(i).SetFocus
Exit Sub
End If
Next
End If
End Sub
```

```

        End If
    Next

    .tabSocial.Tab = 8
    Call CountScore
    Call CountGoals
    txtsocial(11).SetFocus
End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboAgree_Validate(Index As Integer, Cancel As Boolean)
    If Index = 0 Or Index = 1 Or Index = 2 Or Index = 3 Then
        If cboRecommend(0).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 4 Then
        If cboRecommend(1).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 5 Or Index = 6 Or Index = 7 Or Index = 8 Or Index = 9 Then
        If cboRecommend(2).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 10 Or Index = 11 Then
        If cboRecommend(3).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 12 Or Index = 13 Or Index = 14 Or Index = 15 Then
        If cboRecommend(4).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 16 Or Index = 17 Then
        If cboRecommend(5).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 18 Or Index = 19 Or Index = 20 Then
        If cboRecommend(6).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 21 Or Index = 22 Then
        If cboRecommend(7).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 23 Or Index = 24 Or Index = 25 Then
        If cboRecommend(8).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 26 Or Index = 27 Then
        If cboRecommend(9).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    ElseIf Index = 28 Then
        If cboRecommend(10).Text <> "Yes" Then cboAgree(Index).ListIndex = -1
    End If
End Sub

Private Sub cboDifficulty_Click(Index As Integer)

    If IsArrowKey > 0 Then Exit Sub

    If Index = 1 Or Index = 2 Then
        If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(0).Value = 1
    ElseIf Index = 3 Or Index = 4 Then
        If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then
            chkRecommendation(1).Value = 1
            chkRecommendation(2).Value = 1
        End If
    ElseIf Index = 5 Then
        If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(3).Value = 1
    ElseIf Index = 6 Then
        If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(4).Value = 1
    ElseIf Index = 7 Then
        If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(5).Value = 1
    ElseIf Index = 8 Then
        If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Or cboDifficulty(Index).ListIndex = 2 Then
            chkRecommendation(8).Value = 1
        Else
            txtNarrative(6).SetFocus
        End If
    ElseIf Index = 9 Or Index = 10 Or Index = 11 Then
        If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(8).Value = 1
    End Sub

```

```
ndation(6).Value = 1
End If
End Sub
```

```
Private Sub cboDifficulty_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = vbKeyUp Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyDown Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyRight Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyLeft Then
        IsArrowKey = KeyCode
    End If
End Sub
```

```
Private Sub cboDifficulty_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = vbKeyBack Then
        cboDifficulty(Index).ListIndex = 1
    End If
End Sub
```

```
Private Sub cboDifficulty_LostFocus(Index As Integer)
```

```
    If CheckForTab Then
        If Index = 1 Or Index = 2 Then
            If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(0).Value = 1
        ElseIf Index = 3 Or Index = 4 Then
            If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then
                chkRecommendation(1).Value = 1
                chkRecommendation(2).Value = 1
            End If
        ElseIf Index = 5 Then
            If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(3).Value = 1
        ElseIf Index = 6 Then
            If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(4).Value = 1
        ElseIf Index = 7 Then
            If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(5).Value = 1
        ElseIf Index = 8 Then
            If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Or cboDifficulty(Index).ListIndex = 2 Then
                chkRecommendation(8).Value = 1
            Else
                txtNarrative(6).SetFocus
            End If
        ElseIf Index = 9 Or Index = 10 Or Index = 11 Then
            If cboDifficulty(Index).ListIndex = 0 Or cboDifficulty(Index).ListIndex = 1 Then chkRecommendation(6).Value = 1
        End If
    End If

    If IsArrowKey > 0 Then IsArrowKey = 0
End Sub
```

```
Private Sub cboInformation_Click(Index As Integer)
```

```
    If IsArrowKey > 0 Then Exit Sub
```

```
    If Index = 4 Then
        If cboInformation(Index).ListIndex <> 4 Then cboYesNo(6).SetFocus
    ElseIf Index = 6 Then
        If cboInformation(Index).ListIndex <> 3 Then cboInformation(7).SetFocus
    ElseIf Index = 9 Then
        If cboInformation(Index).ListIndex <> 4 Then txtNarrative(4).SetFocus
    End If
```



```
ElseIf Index = 10 Then
    If cboInformation(Index).ListIndex <> 4 Then txtNarrative(5).SetFocus
ElseIf Index = 15 Then
    If cboInformation(15).ListIndex = 0 Or cboInformation(15).ListIndex = 1 Or cboInformation(15)
ListIndex = 2 Then
        chkRecommendation(9).Value = 1
    End If
ElseIf Index = 17 Then
    tabSocial.Tab = 9
End If
End Sub

Private Sub cboInformation_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub

Private Sub cboInformation_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboInformation(Index).ListIndex = -1
End If
End Sub

Private Sub cboInformation_LostFocus(Index As Integer)

If CheckForTab Then
    If Index = 4 Then
        If cboInformation(Index).ListIndex <> 4 Then cboYesNo(6).SetFocus
    ElseIf Index = 6 Then
        If cboInformation(Index).ListIndex <> 3 Then cboInformation(7).SetFocus
    ElseIf Index = 9 Then
        If cboInformation(Index).ListIndex <> 4 Then txtNarrative(4).SetFocus
    ElseIf Index = 10 Then
        If cboInformation(Index).ListIndex <> 4 Then txtNarrative(5).SetFocus
    ElseIf Index = 15 Then
        If cboInformation(15).ListIndex = 0 Or cboInformation(15).ListIndex = 1 Or cboInformation
5).ListIndex = 2 Then
            chkRecommendation(9).Value = 1
        End If
    ElseIf Index = 17 Then
        If CheckForTab Then
            tabSocial.Tab = 9
        End If
    End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboRating_Click(Index As Integer)
Dim i As Integer

If IsArrowKey > 0 Then Exit Sub

If TriggerPosition >= 10 Then
    tabSocial.Tab = 7
Else
    For i = TriggerPosition + 1 To chkTrigger.Count - 1
        If chkTrigger(i).Value = 1 Then
            TriggerPosition = i
            Call GetAssessmentTab
        End If
    Next i
End If
End Sub
```

```
Exit Sub
End If
Next

tabSocial.Tab = 7
End If
End Sub

Private Sub cboRating_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub

Private Sub cboRating_LostFocus(Index As Integer)
Dim i As Integer

If CheckForTab Then
    If TriggerPosition >= 10 Then
        tabSocial.Tab = 7
    Else
        For i = TriggerPosition + 1 To chkTrigger.Count - 1
            If chkTrigger(i).Value = 1 Then
                TriggerPosition = i
                Call GetAssessmentTab
                Exit Sub
            End If
        Next

        tabSocial.Tab = 7
    End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboRecommend_Click(Index As Integer)
Dim i As Integer

If IsArrowKey > 0 Then Exit Sub

If cboRecommend(Index).ListIndex = 1 Then
    For i = GoalPosition + 1 To chkRecommendation.Count - 1
        If chkRecommendation(i).Value = 1 Then
            GoalPosition = i
            If i > 4 Then tabSocial.Tab = 8
            chkRecommendation(i).SetFocus
            Exit Sub
        End If
    Next

    tabSocial.Tab = 8
    Call CountScore
    Call CountGoals
End If
End If
End Sub

Private Sub cboRecommend_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyUp Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyDown Then
    IsArrowKey = KeyCode
```

```
ElseIf KeyCode = vbKeyRight Then
    IsArrowKey = KeyCode
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub
```

```
Private Sub cboRecommend_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = vbKeyBack Then
        cboRecommend(Index).ListIndex = 1
    End If
End Sub
```

```
Private Sub cboRecommend_LostFocus(Index As Integer)
    Dim i As Integer
```

```
    If CheckForTab Then
        If cboRecommend(Index).ListIndex = 1 Then
            For i = GoalPosition + 1 To chkRecommendation.Count - 1
                If chkRecommendation(i).Value = 1 Then
                    GoalPosition = i
                    If i > 4 Then tabSocial.Tab = 8
                    chkRecommendation(i).SetFocus
                    If IsArrowKey > 0 Then IsArrowKey = 0
                    Exit Sub
                End If
            Next
            tabSocial.Tab = 8
            Call CountScore
            Call CountGoals
        End If
    End If
End Sub
```

```
    If IsArrowKey > 0 Then IsArrowKey = 0
End Sub
```

```
Private Sub cboRecommend_Validate(Index As Integer, Cancel As Boolean)
    If chkRecommendation(Index).Value <> 1 Then cboRecommend(Index).ListIndex = -1
End Sub
```

```
Private Sub cboYesNo_Click(Index As Integer)
```

```
    If IsArrowKey > 0 Then Exit Sub

    If Index = 5 Then
        If cboYesNo(Index).ListIndex = 1 Then chkRecommendation(0).Value = 1
    ElseIf Index = 7 Then
        If cboYesNo(Index).ListIndex = 1 Then cboYesNo(8).SetFocus
    ElseIf Index = 8 Then
        If cboYesNo(Index).ListIndex = 1 Then cboYesNo(9).SetFocus
    ElseIf Index = 9 Then
        If cboYesNo(Index).ListIndex = 1 Then txtNarrative(3).SetFocus
    ElseIf Index = 10 Then
        If cboYesNo(Index).ListIndex = 1 Then cboInformation(7).SetFocus
    ElseIf Index = 11 Then
        If cboYesNo(Index).ListIndex = 1 Then cboYesNo(12).SetFocus
    ElseIf Index = 12 Then
        If cboYesNo(Index).ListIndex = 1 Then txtNarrative(3).SetFocus
    ElseIf Index = 13 Then
        If cboYesNo(Index).ListIndex = 1 Then cboYesNo(14).SetFocus
    ElseIf Index = 14 Then
        If cboYesNo(Index).ListIndex = 0 Then chkRecommendation(8).Value = 1
    ElseIf Index = 15 Then
        If cboYesNo(Index).ListIndex = 1 Then cboYesNo(16).SetFocus
    ElseIf Index = 16 Then
        If cboYesNo(Index).ListIndex = 0 Then
            chkRecommendation(6).Value = 1
        Else

```

```
        cboYesNo(17).SetFocus
    End If
ElseIf Index = 17 Then
    If cboYesNo(Index).ListIndex = 0 Then
        chkRecommendation(6).Value = 1
    Else
        txtNarrative(7).SetFocus
    End If
ElseIf Index = 18 Then
    If cboYesNo(Index).ListIndex = 1 Then cboYesNo(19).SetFocus
ElseIf Index = 24 Then
    If cboYesNo(Index).ListIndex = 1 Then cboInformation(10).SetFocus
End If
End Sub

Private Sub cboYesNo_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyUp Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyDown Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyRight Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyLeft Then
        IsArrowKey = KeyCode
    End If
End Sub

Private Sub cboYesNo_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboYesNo(Index).ListIndex = -1
    End If
End Sub

Private Sub cboYesNo_LostFocus(Index As Integer)
    If CheckForTab Then
        If Index = 5 Then
            If cboYesNo(Index).ListIndex = 1 Then chkRecommendation(0).Value = 1
        ElseIf Index = 7 Then
            If cboYesNo(Index).ListIndex = 1 Then cboYesNo(8).SetFocus
        ElseIf Index = 8 Then
            If cboYesNo(Index).ListIndex = 1 Then cboYesNo(9).SetFocus
        ElseIf Index = 9 Then
            If cboYesNo(Index).ListIndex = 1 Then txtNarrative(3).SetFocus
        ElseIf Index = 10 Then
            If cboYesNo(Index).ListIndex = 1 Then cboInformation(7).SetFocus
        ElseIf Index = 11 Then
            If cboYesNo(Index).ListIndex = 1 Then cboYesNo(12).SetFocus
        ElseIf Index = 12 Then
            If cboYesNo(Index).ListIndex = 1 Then txtNarrative(3).SetFocus
        ElseIf Index = 13 Then
            If cboYesNo(Index).ListIndex = 1 Then cboYesNo(14).SetFocus
        ElseIf Index = 14 Then
            If cboYesNo(Index).ListIndex = 0 Then chkRecommendation(8).Value = 1
        ElseIf Index = 15 Then
            If cboYesNo(Index).ListIndex = 1 Then cboYesNo(16).SetFocus
        ElseIf Index = 16 Then
            If cboYesNo(Index).ListIndex = 0 Then
                chkRecommendation(6).Value = 1
            Else
                cboYesNo(17).SetFocus
            End If
        ElseIf Index = 17 Then
            If cboYesNo(Index).ListIndex = 0 Then
                chkRecommendation(6).Value = 1
            Else
                txtNarrative(7).SetFocus
            End If
        End If
    End If
End Sub
```

```

ElseIf Index = 18 Then
    If cboYesNo(Index).ListIndex = 1 Then cboYesNo(19).SetFocus
ElseIf Index = 24 Then
    If cboYesNo(Index).ListIndex = 1 Then cboInformation(10).SetFocus
End If
End If

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cmdAssess_Click()

Call GetAssessmentTab
End Sub

Private Sub cmdAssess_LostFocus()

If CheckForTab Then
    Call GetAssessmentTab
End If
End Sub

Private Sub cmdClear_Click()

IsArrowKey = 20
IsNewRecord = True
clearData lstBenefits, False
clearData cboStaffs, False
clearData txtsocial, True
clearData txtNarrative, True
clearData chkGoals, True
clearData cboYesNo, True
clearData cboAgree, True
clearData chkTrigger, True
clearData cboDifficulty, True
clearData cboRating, True
clearData chkRecommendation, True
clearData cboRecommend, True
clearData cboInformation, True

rst.Open "Select BiggestProblem, Crisis, Crises From Registration Where ConsumerID = " & Custom
ID & _
    " And Agency = " & AgencyCode & " And CaseNumber = " & CaseNumber & _
    " Order by FinishDate", cnn, adOpenForwardOnly, adLockReadOnly
LoadDataToControls rst, cboInformation(0), False
If rst![Crisis] = "1" Then LoadDataToControls rst, lstCrises, False
rst.Close

rst.Open "Select Age, AgeAtOnset, Diabetes, TroubleDiabetes, Hearing, MentionHearing, CheckHeari
, Abuse, " & _
    " Confusion, Delay, Elderly, Frail, Housing, Legal, Neglect, NoPayer, NoSupportive, NoI
From" & _
    " Triage Where ConsumerID = " & CustomerID & " And Agency = " & AgencyCode & _
    " And CaseNumber = " & CaseNumber & " Order by FinishDate ASC", cnn, _
    adOpenForwardOnly, adLockReadOnly
With rst
    .MoveFirst
    If IsNumeric(Trim(![Age])) Then txtsocial(17) = CLng(![Age])
    If IsNumeric(Trim(![AgeAtOnset])) Then txtsocial(16) = ![AgeAtOnset]
    LoadDataToControls rst, cboYesNo(0), False
    LoadDataToControls rst, cboYesNo(1), False
    LoadDataToControls rst, cboYesNo(2), False
    LoadDataToControls rst, cboYesNo(3), False
    LoadDataToControls rst, cboInformation(1), False
    LoadDataToControls rst, chkTrigger, True
End With
rst.Close

If Not rstSocial.BOF Then rstSocial.MoveFirst
txtCurrentRecord = ""

```

```
cmdPrint.Enabled = False
cmdSave.Enabled = True
tabSocial.Tab = 0
IsArrowKey = 0
End Sub
```

```
Private Sub cmdClose_Click()
Unload Me
End Sub
```

```
Private Sub cmdHistorical_Click()
IsVisualInfoClicked = True
frmEyeData.Show vbModal
End Sub
```

```
Private Sub cmdNext_Click()
```

```
If tabSocial.Tab = 9 Then
MsgBox "It is last page for Assessment already!", vbInformation + vbOKOnly, "Next Page"
Else
tabSocial.Tab = tabSocial.Tab + 1
frmSocialWork.Caption = getTabCaption(tabSocial.Tab)
End If
End Sub
```

```
Private Sub cmdNExtRecord_Click()
```

```
With rstSocial
```

```
.MoveNext
If .AbsolutePosition = 1 Then
cmdPreviousRecord.Enabled = False
cmdNExtRecord.Enabled = True
ElseIf .AbsolutePosition < .RecordCount Then
cmdPreviousRecord.Enabled = True
cmdNExtRecord.Enabled = True
ElseIf .RecordCount = .AbsolutePosition Then
cmdPreviousRecord.Enabled = True
cmdNExtRecord.Enabled = False
End If
```

```
If IsNumeric(Trim(![Age])) Then
txtsocial(17) = CLng(![Age])
Call DevelopmentDelayGUI
End If
```

```
txtCurrentRecord = .AbsolutePosition
txtSequencyNo = ![AssessmentSequence]
LoadDataToControls rstSocial, lstCrises, False
LoadDataToControls rstSocial, lstBenefits, False
LoadDataToControls rstSocial, cboStaffs, False
LoadDataToControls rstSocial, txtsocial, True
LoadDataToControls rstSocial, txtNarrative, True
LoadDataToControls rstSocial, chkGoals, True
LoadDataToControls rstSocial, cboYesNo, True
LoadDataToControls rstSocial, cboAgree, True
LoadDataToControls rstSocial, chkTrigger, True
LoadDataToControls rstSocial, cboDifficulty, True
LoadDataToControls rstSocial, cboRating, True
LoadDataToControls rstSocial, chkRecommendation, True
LoadDataToControls rstSocial, cboRecommend, True
LoadDataToControls rstSocial, cboInformation, True
If ![FinishDate] = "12:00:00 PM" Then
cmdSave.Enabled = True
Else
cmdSave.Enabled = False
End If
End With
```

```
tabSocial.Tab = 0
txtCurrentRecord.SetFocus
```

End Sub

Private Sub cmdPrevious\_Click()

If tabSocial.Tab = 0 Then

MsgBox "It is first page for Historical Information already!", vbInformation + vbOKOnly, "P  
iouse Page"

Else

tabSocial.Tab = tabSocial.Tab - 1

frmSocialWork.Caption = getTabCaption(tabSocial.Tab)

End If

End Sub

Private Sub cmdPreviousRecord\_Click()

With rstSocial

.MovePrevious.

If .AbsolutePosition = 1 Then

cmdPreviousRecord.Enabled = False

cmdNextRecord.Enabled = True

ElseIf .AbsolutePosition < .RecordCount Then

cmdPreviousRecord.Enabled = True

cmdNextRecord.Enabled = True

ElseIf .RecordCount = .AbsolutePosition Then

cmdPreviousRecord.Enabled = True

cmdNextRecord.Enabled = False

End If

If ![FinishDate] = "12:00:00 PM" Then

cmdSave.Enabled = True

Else

cmdSave.Enabled = False

End If

If IsNumeric(Trim(![Age])) Then

txtsocial(17) = CLng(![Age])

Call DevelopmentDelayGUI

End If

txtCurrentRecord = .AbsolutePosition

txtSequencyNo = ![AssessmentSequence]

LoadDataToControls rstSocial, lstCrises, False

LoadDataToControls rstSocial, lstBenefits, False

LoadDataToControls rstSocial, cboStaffs, False

LoadDataToControls rstSocial, txtsocial, True

LoadDataToControls rstSocial, txtNarrative, True

LoadDataToControls rstSocial, chkGoals, True

LoadDataToControls rstSocial, cboYesNo, True

LoadDataToControls rstSocial, cboAgree, True

LoadDataToControls rstSocial, chkTrigger, True

LoadDataToControls rstSocial, cboDifficulty, True

LoadDataToControls rstSocial, cboRating, True

LoadDataToControls rstSocial, chkRecommendation, True

LoadDataToControls rstSocial, cboRecommend, True

LoadDataToControls rstSocial, cboInformation, True

End With

tabSocial.Tab = 0

txtCurrentRecord.SetFocus

End Sub

Private Sub cmdPrint\_Click()

Dim i As Integer

If Printers.Count = 0 Then

MsgBox "No printer found associating with this computer. Please add a printer to it and ther  
ou can print out the Service agreement! ", , \_

vbInformation + vbOKOnly, "Printer"

Else

If rst.State = adStateOpen Then rst.Close

```

rst.Open "Select * From SocialWorkAgreement Where AssessmentSequence = " & txtSequencyNo, c
adOpenDynamic, adLockOptimistic
If Not rst.BOF Then ' if there an existing report, delete it then create a new report
rst.Delete
End If
rst.Close

```

```

If rst.State = adStateOpen Then rst.Close
rst.Open "Select * from SocialWorkAgreement Where AssessmentSequence = " & txtSequencyNo, c
adOpenKeyset, adLockOptimistic
With rst

```

```

If .BOF And .EOF Then
.AddNew
![CaseNumber] = txtCaseNumber.
![Episode] = txtEpisode
![AssessmentSequence] = txtSequencyNo
![ConsumerName] = txtConsumerName
![AssessmentDate] = txtsocial(13)
![StaffName] = cboStaffs.Text
For i = 0 To chkGoals.Count - 1
If chkGoals(i).Value = 0 Then
.Fields(chkGoals(i).Tag) = ""
Else
.Fields(chkGoals(i).Tag) = "Yes"
End If
Next

```

```

For i = 20 To 23
.Fields(cboYesNo(i).Tag) = cboYesNo(i).Text
Next

```

```

For i = 18 To 21
.Fields(cboInformation(i).Tag) = cboInformation(i).Text
Next

```

```

.Fields("Comments") = txtsocial(15).Text
.Update
End If

```

```

End With
Set SocialWorkAgreement.DataSource = rst
SocialWorkAgreement.PrintReport

```

```

End If
End Sub

```

```

Private Sub cmdRecommendation_Click()
Dim response As Integer

```

```

response = MsgBox("Based on what you assessed on social issues, We recommend the following goal
)." & _
" Ask Consumer if he or she agrees to the recommendations. If Yes, the consu
must" & _
" agree at least to one of the Objectives for the Goal.", vbOKOnly, _
"Recommended Goal")

```

```

End Sub

```

```

Private Sub cmdRecommendation_LostFocus()
Dim i As Integer

```

```

If CheckForTab Then
For i = 0 To chkRecommendation.Count - 1
If chkRecommendation(i).Value = 1 Then
GoalPosition = i
If i <= 4 Then
chkRecommendation(i).SetFocus
Exit Sub
Else
tabSocial.Tab = 8
chkRecommendation(i).SetFocus
Exit Sub

```



```

        End If
    End If
Next

    tabSocial.Tab = 8
    txtsocial(11).SetFocus
End If
End Sub

Private Sub cmdSave_Click()
    Dim i As Integer
    Dim colSocial As New Collection

    On Error GoTo errSave

    colSocial.Add lstCrises
    colSocial.Add lstBenefits
    colSocial.Add cboStaffs
    colSocial.Add txtsocial
    colSocial.Add txtNarrative
    colSocial.Add chkGoals
    colSocial.Add cboYesNo
    colSocial.Add cboAgree
    colSocial.Add chkTrigger
    colSocial.Add cboDifficulty
    colSocial.Add cboRating
    colSocial.Add chkRecommendation
    colSocial.Add cboRecommend
    colSocial.Add cboInformation

    If SaveDataToDB(frmSocialWork, rstSocial, colSocial, "010", False) Then 'Call public fu
        ction to save data to DB
        MsgBox "Successfully saved the Assessment information!", vbOKOnly, "Save"

        If IsDate(txtsocial(13)) Then
            cnn.Execute "Update Interventions Set PreConsumerScore = ' " & txtConsumerScore & "', PrePr
viderScore = ' " & _
                txtProviderScore & " ' Where ConsumerID = " & CustomerID & " And Agency = ' " & AgencyCod
& _
                " ' And Episode = " & Episode & " And CaseNumber = ' " & CaseNumber & _
                " ' And InterventionID = '010' And (EndDate = #12:00:00 PM# or EndDate IS NULL)"

            cnn.Execute "UPDATE SocialAssessment AS a INNER JOIN Interventions AS b ON (a.CaseNumber =
b.CaseNumber) " & _
                " AND (a.Episode = b.Episode) AND (a.ConsumerID = b.ConsumerID) SET a.InterventionSeque
ce = b.InterventionSequence" & _
                " WHERE b.InterventionID = '010' and b.CaseNumber = ' " & CaseNumber & " ' AND (b.EndDate
= #12:00:00PM# OR b.EndDate IS NULL)" & _
                " AND a.InterventionSequence = 0"

        End If

        If IsNewRecord Then
            rstSocial.Requery
            txtSequencyNo = rstSocial![AssessmentSequence]
        End If

        IsNewRecord = False
        Set colSocial = Nothing
    End If
Exit Sub

errSave:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub

Private Sub cmdTrigger_Click()
    Dim response As Integer

    response = MsgBox("From Triage, following checked trigger(s) promoted a Social Service assessmen
", _

```

```

        vbOKOnly, "Trigger")
End Sub

Private Sub cmdTrigger_LostFocus()
Dim i As Integer

If CheckForTab Then
    For i = 0 To chkTrigger.Count - 1
        If chkTrigger(i).Value = 1 Then
            TriggerPosition = i
            chkTrigger(i).SetFocus
            Exit Sub
        End If
    Next
End If
End Sub

Private Sub Combo1_LostFocus(Index As Integer)
If Index = 0 Then
    tabSocial.Tab = 1
ElseIf Index = 1 Then
    tabSocial.Tab = 9
End If
End Sub

Private Sub Form_Activate()
Dim j As Integer

On Error GoTo Errhandler

If IsVisualInfoClicked Then Exit Sub

txtConsumerName = FirstName & " " & LastName
txtPhone = PhoneNo

txtAddress = Address
txtCaseNumber = CaseNumber
txtEpisode = Episode
txtIntervention = "Social Service"

LoadControlsItems cboStaffs, False           'Add every description from database to combobox or listbox
IsArrowKey = 20                             'Avoid Click event when load data from database

Set rst = New ADODB.Recordset
rst.CursorLocation = adUseClient

Set rstSocial = New ADODB.Recordset
rstSocial.CursorLocation = adUseClient
rstSocial.Open "Select * From SocialAssessment Where ConsumerID = " & CustomerID & _
    " And Agency = " & AgencyCode & " And Episode = " & Episode & " And CaseNumber = " & _
    CaseNumber & " And InterventionID = '010' Order By AssessmentSequence DESC", cnn, _
    adOpenKeyset, adLockOptimistic

With rstSocial
    txtRecordNo = .RecordCount
    If Not (.EOF And .BOF) Then               'Have record for this client and load it to form
        If IsNumeric(Trim(![Age])) Then
            txtsocial(17) = CLng(![Age])
            Call DevelopmentDelayGUI
        End If

        txtCurrentRecord = .AbsolutePosition
        txtSequencyNo = ![AssessmentSequence]
        LoadDataToControls rstSocial, lstCrises, False
        LoadDataToControls rstSocial, lstBenefits, False
        LoadDataToControls rstSocial, cboStaffs, False
        LoadDataToControls rstSocial, txtsocial, True
        LoadDataToControls rstSocial, txtNarrative, True
        LoadDataToControls rstSocial, chkGoals, True
    End With

```

```

LoadDataToControls rstSocial, cboYesNo, True
LoadDataToControls rstSocial, cboAgree, True
LoadDataToControls rstSocial, chkTrigger, True
LoadDataToControls rstSocial, cboDifficulty, True
LoadDataToControls rstSocial, cboRating, True
LoadDataToControls rstSocial, chkRecommendation, True
LoadDataToControls rstSocial, cboRecommend, True
LoadDataToControls rstSocial, cboInformation, True

```

```

If .RecordCount > .AbsolutePosition Then cmdNextRecord.Enabled = True

```

```

If ![FinishDate] <> "12:00:00 PM" And IsDate(![FinishDate]) Then
    cmdSave.Enabled = False
    cmdPrint.Enabled = True
    cmdClear.Enabled = True
End If

```

```

Else

```

```

    rst.Open "Select BiggestProblem, Crisis, Crises From Registration Where ConsumerID = " &
stomerID, _
    cnn, adOpenForwardOnly, adLockReadOnly
    LoadDataToControls rst, cboInformation(0), False
    If rst![Crisis] = "1" Then LoadDataToControls rst, lstCrises, False
    rst.Close

```

```

    rst.Open "Select Age, AgeAtOnset, Diabetes, TroubleDiabetes, Hearing, MentionHearing, Cl
Hearing, Abuse, " & _
    "Confusion, Delay, Elderly, Frail, Housing, Legal, Neglect, NoPayer, NoSupportive, No
From" & _

```

```

    "Triage Where ConsumerID = " & CustomerID & " And Agency = " & AgencyCode & " And
eNumber = " & _

```

```

    CaseNumber & " And Episode = " & Episode, cnn, adOpenForwardOnly, adLockReadOnly
With rst

```

```

    .MoveFirst

```

```

    If IsNumeric(Trim(![Age])) Then

```

```

        txtsocial(17) = CLng(![Age])

```

```

        Call DevelopmentDelayGUI

```

```

    End If

```

```

    If IsNumeric(Trim(![AgeAtOnset])) Then txtsocial(16) = ![AgeAtOnset]

```

```

    LoadDataToControls rst, cboYesNo(0), False

```

```

    LoadDataToControls rst, cboYesNo(1), False

```

```

    LoadDataToControls rst, cboYesNo(2), False

```

```

    LoadDataToControls rst, cboYesNo(3), False

```

```

    LoadDataToControls rst, cboInformation(1), False

```

```

    LoadDataToControls rst, chkTrigger, True

```

```

End With

```

```

rst.Close

```

```

cmdNextRecord.Enabled = False

```

```

cmdPreviousRecord.Enabled = False

```

```

txtCurrentRecord = 0

```

```

End If

```

```

End With

```

```

IsArrowKey = 0

```

```

chkTrigger(3).Value = 1

```

```

tabSocial.Tab = 0

```

```

txtConsumerName.SetFocus

```

```

Exit Sub

```

```

Errhandler:

```

```

MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"

```

```

End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)

```

```

Dim response As Integer

```

```

If IsFileDateChecked Then

```

```

    If rstSocial.State = adStateOpen Then rstSocial.Close

```

```

    Set rstSocial = Nothing

```

```
tabSocial.Tab = 0
Set frmSocialWork = Nothing
Else
    If Trim(txtsocial(13)) = "" And cmdSave.Enabled Then
        response = MsgBox("Have you finished the Social Service Assessment ?", vbQuestion + vbYe
        , "File Date")
        If response = vbYes Then
            MsgBox "Please enter the File Date !", vbInformation + vbOKOnly, "File Date"
            txtsocial(13).SetFocus
            Cancel = True
        Else
            If rstSocial.State = adStateOpen Then rstSocial.Close
            Set rstSocial = Nothing
            tabSocial.Tab = 0
            Set frmSocialWork = Nothing
        End If
    ElseIf IsDate(Trim(txtsocial(13))) And cmdSave.Enabled Then
        response = MsgBox("Are you sure you finished the Social Service Assessment?", vbQuestion
        vbYesNo, "File Date")
        If response = vbYes Then
            MsgBox "Please use Save button to save the assessment information", vbInformation +
            KOnly, "File Date"
            cmdSave.SetFocus
        ElseIf cmdSave.Enabled = True Then
            txtsocial(13) = ""
        End If
    ElseIf IsDate(Trim(txtsocial(13))) And Not cmdSave.Enabled Then
        If rstSocial.State = adStateOpen Then rstSocial.Close
        Set rstSocial = Nothing
        tabSocial.Tab = 0
        Set frmSocialWork = Nothing
    End If
End If
End Sub

Private Sub lstBenefits_KeyUp(KeyCode As Integer, Shift As Integer)
Dim i As Integer

If KeyCode = vbKeyBack Then
    If lstBenefits.SelCount = 1 Then
        lstBenefits.Selected(lstBenefits.ListIndex) = False
    ElseIf lstBenefits.ListCount > 1 Then
        For i = lstBenefits.ListCount - 1 To 0 Step -1
            If lstBenefits.Selected(i) Then
                lstBenefits.Selected(i) = False
            End If
        Next
    End If
End If
End Sub

Private Sub lstBenefits_LostFocus()

If Not lstBenefits.Selected(5) Then txtsocial(14).SetFocus
End Sub

Private Sub tabSocial_Click(PreviousTab As Integer)
Dim j As Integer

If tabSocial.Tab = 0 Then
    cboStaffs.SetFocus
    TriggerPosition = 0
    frmSocialWork.Caption = "A. Historical Information"
ElseIf tabSocial.Tab = 1 Then
    frmSocialWork.Caption = "B. No Support Network/Isolation"
ElseIf tabSocial.Tab = 2 Then
    cboDifficulty(3).SetFocus
    frmSocialWork.Caption = "C. Frailty"
ElseIf tabSocial.Tab = 3 Then
    If Not IsHousingIssues Then
```

```

frmSocialWork.Caption = "D. Development Delay"
If CInt(txtsocial(17)) < 21 Then
    cboYesNo(7).SetFocus
Else
    cboYesNo(10).SetFocus
End If
End If
ElseIf tabSocial.Tab = 4 Then
    frmSocialWork.Caption = "E. Legal Issues"
ElseIf tabSocial.Tab = 5 Then
    cboDifficulty(9).SetFocus
    frmSocialWork.Caption = "F. Abuse/Neglect"
ElseIf tabSocial.Tab = 6 Then
    frmSocialWork.Caption = "G. No Payer Available"
ElseIf tabSocial.Tab = 7 Then
    cmdRecommendation.SetFocus
    frmSocialWork.Caption = "H. Recommended Goals"
ElseIf tabSocial.Tab = 8 Then
    frmSocialWork.Caption = "I. Recommended Goals"
ElseIf tabSocial.Tab = 9 Then
    frmSocialWork.Caption = "J. Social Service Agreement"
    cmdGoal.SetFocus
CountGoals
End If

End Sub

Private Function getTabCaption(tabIndex As Integer) As String
Select Case tabIndex
    Case 0
        getTabCaption = "A. Historical Information"
    Case 1
        getTabCaption = "B. No Support Network/Isolation"
    Case 2
        getTabCaption = "C. Frailty"
    Case 3
        getTabCaption = "D. Development Delay"
    Case 4
        getTabCaption = "E. Legal Issues"
    Case 5
        getTabCaption = "F. Abuse/Neglect"
    Case 6
        getTabCaption = "G. No Payer Available"
    Case 7
        getTabCaption = "H. Recommended Goals"
    Case 8
        getTabCaption = "I. Recommended Goals"
    Case 9
        getTabCaption = "J. Social Services Agreement"
End Select
End Function

Private Sub CountScore()
Dim i As Integer
Dim tmpValue As Integer
Dim taskA As Long
Dim taskB As Long
Dim avgScore As Single

For i = 0 To cboDifficulty.Count - 1 'count the score for task A
    If cboDifficulty(i).ListIndex = -1 Then
        taskA = taskA + 4
    Else
        tmpValue = cboDifficulty(i).ItemData(cboDifficulty(i).ListIndex)
        taskA = taskA + tmpValue
    End If
Next

tmpValue = 0

```

```
For i = 0 To cboRating.Count - 1
    If cboRating(i).ListIndex = -1 Then 'user does'nt select one, count 10 for this item
        taskB = taskB + 10
    Else
        tmpValue = cboRating(i).ItemData(cboRating(i).ListIndex)
        taskB = taskB + tmpValue
    End If
Next
```

```
txtProviderScore = taskB * 1.1
txtConsumerScore = taskA * 1.7
avgScore = ((taskA * 1.7) + (taskB * 1.1)) / 2
txtsocial(11) = CLng(avgScore)
txtsocial(18) = Format(taskA * 1.7, "##")
txtsocial(19) = Format(taskB * 1.1, "##")
txtsocial(20) = Format(avgScore, "##")
```

```
Select Case CLng(avgScore)
    Case 0 To 60:    txtsocial(12) = "Level 3"
    Case 61 To 80:  txtsocial(12) = "Level 2"
    Case 81 To 90:  txtsocial(12) = "Level 1"
    Case 91 To 100: txtsocial(12) = "No Service Needed"
End Select
End Sub
```

```
Private Sub txtNarrative_LostFocus(Index As Integer)
Dim i As Integer
```

```
If CheckForTab Then
    If Index = 3 Then
        If TriggerPosition >= 10 Then
            tabSocial.Tab = 7
            cmdRecommendation.SetFocus
        Else
            For i = TriggerPosition + 1 To chkTrigger.Count - 1
                If chkTrigger(i).Value = 1 Then
                    TriggerPosition = i
                    Call GetAssessmentTab
                    Exit Sub
                End If
            Next
            tabSocial.Tab = 7
            cmdRecommendation.SetFocus
        End If
    End If
End If
End Sub
```

```
Private Sub GetAssessmentTab()
Dim i As Integer, j As Integer
```

```
For i = TriggerPosition To chkTrigger.Count - 1
    If i = 3 Then
        TriggerPosition = i
        tabSocial.Tab = 1
        cboYesNo(4).SetFocus
        Exit For
    End If

    If chkTrigger(i).Value = 1 Then
        If i = 0 Then
            TriggerPosition = i
            tabSocial.Tab = 5
            Exit For
        ElseIf i = 1 Then
            TriggerPosition = i
            tabSocial.Tab = 2
            frmSocialWork.Caption = "Confusion/disorientation"
```

```
        cboDifficulty(5).SetFocus
    Exit For
ElseIf i = 2 Then
    If Not IsHousingIssues Then
        TriggerPosition = i
        tabSocial.Tab = 3
        frmSocialWork.Caption = "Development Delay"
    End If
    Exit For
ElseIf i = 4 Then
    TriggerPosition = i
    tabSocial.Tab = 2
    frmSocialWork.Caption = "Frailty"
    cboDifficulty(3).SetFocus
    Exit For
ElseIf i = 5 Then
    IsHousingIssues = True
    TriggerPosition = i
    tabSocial.Tab = 3
    frmSocialWork.Caption = "Housing Issues"
    cboDifficulty(6).SetFocus
    Exit For
ElseIf i = 6 Then
    TriggerPosition = i
    tabSocial.Tab = 4
    frmSocialWork.Caption = "Legal Issues"
    cboYesNo(24).SetFocus
    Exit For
ElseIf i = 7 Then
    If chkTrigger(0).Value = 0 Then
        TriggerPosition = i
        tabSocial.Tab = 5
        cboDifficulty(9).SetFocus
    End If
    Exit For
ElseIf i = 8 Then
    TriggerPosition = i
    tabSocial.Tab = 6
    frmSocialWork.Caption = "No Payer Available"
    cboInformation(14).SetFocus
    Exit For
ElseIf i = 10 Then
    TriggerPosition = i
    tabSocial.Tab = 6
    frmSocialWork.Caption = "Refuse to pay"
    cboInformation(15).SetFocus
    Exit For
End If
End If
Next
End Sub

Private Sub CountGoals()
    Dim i As Integer

    For i = 0 To cboAgree.Count - 1
        Select Case i
            Case 0 To 3
                If cboAgree(i).Text = "Yes" Then chkGoals(0) = 1
            Case 4
                If cboAgree(i).Text = "Yes" Then chkGoals(1) = 1
            Case 5 To 9
                If cboAgree(i).Text = "Yes" Then chkGoals(2) = 1
            Case 10 To 11
                If cboAgree(i).Text = "Yes" Then chkGoals(3) = 1
            Case 12 To 15
                If cboAgree(i).Text = "Yes" Then chkGoals(4) = 1
            Case 16 To 17
                If cboAgree(i).Text = "Yes" Then chkGoals(5) = 1
            Case 18 To 20
```

```
        If cboAgree(i).Text = "Yes" Then chkGoals(6) = 1
    Case 21 To 22
        If cboAgree(i).Text = "Yes" Then chkGoals(7) = 1
    Case 23 To 25
        If cboAgree(i).Text = "Yes" Then chkGoals(8) = 1
    Case 26 To 27
        If cboAgree(i).Text = "Yes" Then chkGoals(9) = 1
    Case 28
        If cboAgree(i).Text = "Yes" Then chkGoals(10) = 1
    End Select
Next
End Sub

Private Sub txtSocial_LostFocus(Index As Integer)
    If CheckForTab Then
        If Index = 10 Then txtNarrative(9).SetFocus
    End If
End Sub

Private Sub DevelopmentDelayGUI()
    Dim j As Integer

    If Cint(txtsocial(17)) <= 21 Then
        For j = 0 To 5
            lblDelay(j).Visible = True
        Next

        cboYesNo(7).Visible = True
        cboYesNo(8).Visible = True
        cboYesNo(9).Visible = True
        cboInformation(5).Visible = True
        txtsocial(2).Visible = True
        txtsocial(3).Visible = True
        lblDelay(14).Caption = "3. Narrative:"
    Else
        For j = 6 To 13
            lblDelay(j).Visible = True
        Next

        cboYesNo(10).Visible = True
        cboYesNo(11).Visible = True
        cboYesNo(12).Visible = True
        cboInformation(6).Visible = True
        cboInformation(7).Visible = True
        cboInformation(8).Visible = True
        txtsocial(4).Visible = True
        txtsocial(5).Visible = True
    End If
End Sub
```



```

Option Explicit
Private tmpEpisode As Integer
Private chkLocation As Integer
CheckBox
Private IsSighted As Boolean
Private IsShftTab As Boolean
Public IsNewRecord As Boolean
Private ServiceLocation As Integer
Private IsFileDateChecked As Boolean
Private rstTriage As ADODB.Recordset
Private IsArrowKey As Integer

'to remember the location of Service Recommendedat:

Private Sub cboAcceptReject_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
If KeyCode = vbKeyBack Then
    cboAcceptReject(Index).ListIndex = -1
End If
End Sub

Private Sub cboAcceptReject_LostFocus(Index As Integer)
Dim i As Integer
Dim IsRecommended As Boolean
Dim HasOtherReason As Boolean

If CheckForTab And cboAcceptReject(Index).Text = "Rejected" Then
    cboRejectionReason(Index).SetFocus
ElseIf CheckForTab And cboAcceptReject(Index).Text <> "Rejected" Then
    For i = ServiceLocation To chkRecommendation.Count - 1
        If chkRecommendation(i).Value = 1 Then
            ServiceLocation = i
            chkRecommendation(i).SetFocus
            IsRecommended = True
            Exit Sub
        End If
    Next

    If Not IsRecommended Then
        For i = 0 To ServiceLocation
            If cboRejectionReason(i).ListIndex = 5 Then
                txtTriage(10).SetFocus
                HasOtherReason = True
            End If
        Next

        If Not HasOtherReason Then
            tabTriage.Tab = 6
            cmdInstruction.SetFocus
        End If
    End If
ElseIf CheckForShftTab Then
    chkRecommendation(Index).SetFocus
Else
    'If user manually select one "Accepted" combo, its service is recommended
    If cboAcceptReject(Index).Text = "Accepted" And chkRecommendation(Index).Value = 0 Then
        chkRecommendation(Index).Value = 1
    End If
End If
End Sub

Private Sub cboActionStatus_KeyUp(KeyCode As Integer, Shift As Integer)
If KeyCode = vbKeyBack Then
    cboActionStatus.ListIndex = -1
End If
End Sub

Private Sub cboActivity_Click()
If IsArrowKey > 0 Then Exit Sub

If cboActivity.ListIndex = 2 Or cboActivity.ListIndex = 3 Then
    cboYesNo(28).ListIndex = -1

```

---

```
cboFrequency2.SetFocus
```

```
End If
```

```
End Sub
```

```
Private Sub cboActivity_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyUp Then
```

```
    IsArrowKey = KeyCode
```

```
ElseIf KeyCode = vbKeyDown Then
```

```
    IsArrowKey = KeyCode
```

```
ElseIf KeyCode = vbKeyRight Then
```

```
    IsArrowKey = KeyCode
```

```
ElseIf KeyCode = vbKeyLeft Then
```

```
    IsArrowKey = KeyCode
```

```
End If
```

```
End Sub
```

```
Private Sub cboActivity_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyBack Then
```

```
    cboActivity.ListIndex = -1
```

```
End If
```

```
End Sub
```

```
Private Sub cboActivity_LostFocus()
```

```
If CheckForShiftTab Then
```

```
    If Trim(txtTriage(20)) <> "" Then
```

```
        If Cint(Trim(txtTriage(20))) < 18 Then
```

```
            IsShftTab = True
```

```
            tabTriage.Tab = 3
```

```
            cboYesNo(19).SetFocus
```

```
        Else
```

```
            cboYesNo(27).SetFocus
```

```
        End If
```

```
    End If
```

```
ElseIf CheckForTab Then
```

```
    If cboActivity.ListIndex = 2 Or cboActivity.ListIndex = 3 Then cboFrequency2.SetFocus
```

```
End If
```

```
If IsArrowKey > 0 Then IsArrowKey = 0
```

```
End Sub
```

```
Private Sub cboAgree_Click(Index As Integer)
```

```
If IsArrowKey > 0 Then Exit Sub
```

```
If Index = 0 Then
```

```
    If cboAgree(0).Text = "Yes" Then lstRace.SetFocus
```

```
ElseIf Index = 1 Then
```

```
    If cboAgree(0).ListIndex = 0 Then
```

```
        cboAgree(1).ListIndex = -1
```

```
    Else
```

```
        If cboAgree(1).Text = "Yes" Then
```

```
            chkRecommendation(0).Value = 1
```

```
        Else
```

```
            chkRecommendation(0).Value = 0
```

```
        End If
```

```
    End If
```

```
End If
```

```
End Sub
```

```
Private Sub cboAgree_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
If KeyCode = vbKeyUp Then
```

```
    IsArrowKey = KeyCode
```

```
ElseIf KeyCode = vbKeyDown Then
```

```
    IsArrowKey = KeyCode
```

```
ElseIf KeyCode = vbKeyRight Then
```

```
    IsArrowKey = KeyCode
```

```
ElseIf KeyCode = vbKeyLeft Then
    IsArrowKey = KeyCode
End If
End Sub
```

```
Private Sub cboAgree_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = vbKeyBack Then
        cboAgree(Index).ListIndex = -1
    End If
End Sub
```

```
Private Sub cboAgree_LostFocus(Index As Integer)
```

```
    If Index = 0 Then
        If CheckForTab Then
            If cboAgree(0).Text = "Yes" Then lstRace.SetFocus
        End If
    ElseIf Index = 1 Then
        If CheckForTab Then
            If cboAgree(0).ListIndex = 0 Then
                cboAgree(1).ListIndex = -1
            Else
                If cboAgree(1).Text = "Yes" Then
                    chkRecommendation(0).Value = 1
                Else
                    chkRecommendation(0).Value = 0
                End If
            End If
        End If
    End If
End If
```

```
    If IsArrowKey <> 0 Then IsArrowKey = 0
End Sub
```

```
Private Sub cboDenyReason_LostFocus()
```

```
    If CheckForTab Then
        tabTriage.Tab = 6
        cmdInstruction.SetFocus
    End If
End Sub
```

```
Private Sub cboEpisodeOutcome_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = vbKeyBack Then
        cboEpisodeOutcome.ListIndex = -1
    End If
End Sub
```

```
Private Sub cboFrequency2_Click()
```

```
    If IsArrowKey > 0 Then Exit Sub
End Sub
```

```
Private Sub cboFrequency2_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = vbKeyBack Then
        cboFrequency2.ListIndex = -1
    ElseIf KeyCode = vbKeyUp Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyDown Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyRight Then
        IsArrowKey = KeyCode
    ElseIf KeyCode = vbKeyLeft Then
        IsArrowKey = KeyCode
    End If
End Sub
```

```
Private Sub cboFrequency2_LostFocus()
```

```
    If CheckForShiftTab Then
```

```
        If cboActivity.ListIndex = 0 Or cboActivity.ListIndex = 1 Then
```

```
            cboYesNo(28).SetFocus
```

```
        Else
```

```
            cmdFactors.SetFocus
```

```
        End If
```

```
    End If
```

```
    If cboFrequency2.ListIndex = 2 Or cboFrequency2.ListIndex = 3 Then cmdFactors.SetFocus
```

```
    If IsArrowKey > 0 Then IsArrowKey = 0
```

```
End Sub
```

```
Private Sub cboOverallHealth_Click(Index As Integer)
```

```
    If IsArrowKey > 0 Then Exit Sub
```

```
    If Index = 1 Then
```

```
        If cboOverallHealth(1).ListIndex = 0 Or cboOverallHealth(1).ListIndex = 1 Then
```

```
            cboYesNo(7).ListIndex = -1
```

```
            cboYesNo(8).ListIndex = -1
```

```
            cboYesNo(9).SetFocus
```

```
        ElseIf cboOverallHealth(1).ListIndex = 2 Or cboOverallHealth(1).ListIndex = 3 Then
```

```
            cboYesNo(7).SetFocus
```

```
        End If
```

```
    End If
```

```
End Sub
```

```
Private Sub cboOverallHealth_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = vbKeyUp Then
```

```
        IsArrowKey = KeyCode
```

```
    ElseIf KeyCode = vbKeyDown Then
```

```
        IsArrowKey = KeyCode
```

```
    ElseIf KeyCode = vbKeyRight Then
```

```
        IsArrowKey = KeyCode
```

```
    ElseIf KeyCode = vbKeyLeft Then
```

```
        IsArrowKey = KeyCode
```

```
    End If
```

```
End Sub
```

```
Private Sub cboOverallHealth_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
    If KeyCode = vbKeyBack Then
```

```
        cboOverallHealth(Index).ListIndex = -1
```

```
    End If
```

```
End Sub
```

```
Private Sub cboOverallHealth_LostFocus(Index As Integer)
```

```
    If Index = 0 Then
```

```
        If CheckForShiftTab Then
```

```
            IsShiftTab = True
```

```
            tabTriage.Tab = 0
```

```
            If IsSighted Then
```

```
                txtTriage(1).SetFocus
```

```
            ElseIf cboYesNo(2).Text = "Yes" Then
```

```
                cboYesNo(4).SetFocus
```

```
            Else
```

```
                txtTriage(1).SetFocus
```

```
            End If
```

```
        End If
```

```
    ElseIf Index = 1 Then
```

```
        If CheckForShiftTab Then
```

```
            If cboYesNo(5).Text = "No" Then cboYesNo(5).SetFocus
```

```
        ElseIf CheckForTab Then
```

```
            If cboOverallHealth(1).ListIndex = 0 Or cboOverallHealth(1).ListIndex = 1 Then
```

```
                cboYesNo(9).SetFocus
```

```
ElseIf cboOverallHealth(1).ListIndex = 2 Or cboOverallHealth(1).ListIndex = 3 Then
    cboYesNo(7).SetFocus
End If
End If
End Sub

If IsArrowKey > 0 Then IsArrowKey = 0
End Sub

Private Sub cboRejectionReason_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)
    If KeyCode = vbKeyBack Then
        cboRejectionReason(Index).ListIndex = -1
    End If
End Sub

Private Sub cboRejectionReason_LostFocus(Index As Integer)
    Dim i As Integer
    Dim IsRecommended As Boolean
    Dim HasOtherReason As Boolean
    Dim totalRecommendation As Integer
    Dim totalRejection As Integer

    If CheckForTab Then
        For i = ServiceLocation + 1 To chkRecommendation.Count - 1
            If chkRecommendation(i).Value = 1 Then
                ServiceLocation = i
                chkRecommendation(i).SetFocus
                IsRecommended = True
                Exit Sub
            End If
        Next
    End If

    If Not IsRecommended Then
        For i = 0 To ServiceLocation
            If cboRejectionReason(i).ListIndex = 5 Then
                txtTriage(10).SetFocus
                HasOtherReason = True
            End If
        Next

        If Not HasOtherReason And CheckForTab Then
            For i = 0 To chkRecommendation.Count - 1
                If chkRecommendation(i).Value = 1 Then totalRecommendation = totalRecommendation + 1
            Next

            For i = 0 To cboAcceptReject.Count - 1
                If cboAcceptReject(i).Text = "Rejected" Then totalRejection = totalRejection + 1
            Next

            If totalRecommendation = totalRejection Then
                cboDenyReason.SetFocus
            Else
                tabTriage.Tab = 6
                cmdInstruction.SetFocus
            End If
        End If
    End If
End Sub

Private Sub cboYesNo_Click(Index As Integer)
    Dim i As Integer

    If IsArrowKey > 0 Then Exit Sub

    If Index = 0 Then
        If cboYesNo(Index).ListIndex = 0 Then
            cboYesNo(1).ListIndex = -1
            cboYesNo(2).ListIndex = -1
        End If
    End If
End Sub
```

```

    txtTriage(1).SetFocus
End If
ElseIf Index = 1 Then
    If cboYesNo(0).ListIndex = 0 Then
        cboYesNo(Index).ListIndex = -1
    ElseIf cboYesNo(Index).ListIndex = 0 Then
        txtTriage(1).SetFocus
    End If
ElseIf Index = 2 Then
    If cboYesNo(0).ListIndex = 0 Or cboYesNo(1).ListIndex = 0 Then
        cboYesNo(Index).ListIndex = -1
    End If
ElseIf Index = 3 Or Index = 4 Then
    If cboYesNo(2).ListIndex = 1 Then
        cboYesNo(Index).ListIndex = -1
    End If
ElseIf Index = 5 Then
    If cboYesNo(Index).ListIndex <> 0 Then
        cboYesNo(6).ListIndex = -1
        cboOverallHealth(1).SetFocus
    End If
ElseIf Index = 6 Then
    If cboYesNo(5).ListIndex <> 0 Then cboYesNo(Index).ListIndex = -1
    If cboYesNo(Index).Text = "Yes" Then
        chkRecommendation(2).Value = 1      ' If yes to have trouble managing your diabetes,
        chkRecommendation(3).Value = 1      ' recommend Diabetes Management & IDT
    End If
ElseIf Index = 7 Or Index = 8 Then
    If cboOverallHealth(1).ListIndex = 0 Or cboOverallHealth(1).ListIndex = 1 Then
        cboYesNo(Index).ListIndex = -1
    End If
ElseIf Index = 9 Then
    If cboYesNo(9).Text = "Yes" Then
        lstHealthCondition.SetFocus
    Else
        txtTriage(4).SetFocus
    End If
ElseIf Index = 10 Then
    If cboYesNo(Index).ListIndex = 0 Then
        cmdMovement.SetFocus
    If cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value = 1
    End If
ElseIf Index = 11 Then
    If cboYesNo(10).ListIndex = 0 Then
        cboYesNo(11).ListIndex = -1
        cmdMovement.SetFocus
    ElseIf cboYesNo(Index).ListIndex = 0 Then
        If cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value = 1
    End If
ElseIf Index = 12 Then
    If cboYesNo(Index).ListIndex = 0 Then
        If cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value = 1
        chkRecommendation(6).Value = 1
        cmdHousehold.SetFocus
    End If
ElseIf Index = 13 Then
    If cboYesNo(12).ListIndex = 0 Then
        cboYesNo(13).ListIndex = -1
        cmdHousehold.SetFocus
    ElseIf cboYesNo(Index).ListIndex = 0 Then
        If cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value = 1
        chkRecommendation(6).Value = 1
    End If
ElseIf Index = 14 Then
    If cboYesNo(Index).ListIndex = 0 Then
        For i = 0 To 2
            If cboYesNo(i).Text = "Yes" Then chkRecommendation(1).Value = 1
        Next
        chkRecommendation(5).Value = 1

```

```

        tabTriage.Tab = 3
        cmdPsychological.SetFocus
    End If
ElseIf Index = 15 Then
    If cboYesNo(14).ListIndex = 0 Then
        cboYesNo(15).ListIndex = -1
    ElseIf cboYesNo(15).ListIndex = 0 Then
        chkRecommendation(5).Value = 1
        cmdPsychological.SetFocus
    Else
        cboYesNo(16).SetFocus
    End If
    tabTriage.Tab = 3
ElseIf Index = 16 Then
    If cboYesNo(14).ListIndex = 0 Or cboYesNo(15).ListIndex = 0 Then
        cboYesNo(16).ListIndex = -1
    ElseIf cboYesNo(16).ListIndex = 0 Then
        chkRecommendation(5).Value = 1
        cmdPsychological.SetFocus
    Else
        cboYesNo(17).SetFocus
    End If
ElseIf Index = 17 Then
    If cboYesNo(14).ListIndex = 0 Or cboYesNo(15).ListIndex = 0 Or cboYesNo(16).ListIndex = 0
hen
        cboYesNo(17).ListIndex = -1
        cmdPsychological.SetFocus
    ElseIf cboYesNo(Index).ListIndex = 0 Then
        chkRecommendation(5).Value = 1
    End If
ElseIf Index = 18 Then
    If cboYesNo(Index).ListIndex <> -1 Then cboYesNo(19).SetFocus
    If cboYesNo(Index).ListIndex = 0 Then chkRecommendation(4).Value = 1
ElseIf Index = 19 Then
    If cboYesNo(Index).ListIndex <> -1 Then
        If cboYesNo(18).ListIndex = -1 Then
            MsgBox "Please answer Question 1!", vbInformation + vbOKOnly, "Psychological Status"
            cboYesNo(18).SetFocus
        ElseIf cboYesNo(19).ListIndex = 0 Then chkRecommendation(4).Value = 1
        End If
    End If
ElseIf Index = 20 Then
    If cboYesNo(Index).ListIndex = 1 Then
        cboYesNo(23).SetFocus
    Else
        cboYesNo(21).SetFocus
    End If
ElseIf Index = 21 Or Index = 22 Then
    If cboYesNo(20).ListIndex = 1 Then cboYesNo(Index).ListIndex = -1
ElseIf Index = 23 Then
    If cboYesNo(Index).Text = "Yes" Then
        If cboAgree(0).Text = "Yes" Then
            'Yes to like to be employed and is Citizen
            chkRecommendation(9).Value = 1
            cboYesNo(25).SetFocus
        ElseIf cboAgree(0).Text = "No" Then
            'Yes to like be employed and is not citizen
            cboYesNo(24).SetFocus
        End If
    ElseIf cboYesNo(Index).Text = "No" Then
        If Trim(txtTriage(20)) <> "" Then
            If CInt(txtTriage(20)) >= 18 Or CInt(txtTriage(20)) <= 64 Then chkRecommendation(9).Value = 1
e = 1
        End If
    End If
ElseIf Index = 24 Then
    If cboYesNo(Index).Text = "Yes" Then
        chkRecommendation(9).Value = 1
    ElseIf cboYesNo(Index).Text = "No" Then
        cboYesNo(26).SetFocus
    End If
ElseIf Index = 25 Then

```

```
If cboYesNo(Index).Text = "Yes" Then chkRecommendation(10).Value = 1
ElseIf Index = 26 Then      ' Computer Skills
    If cboYesNo(26).Text = "Yes" Then chkRecommendation(8).Value = 1
ElseIf Index = 27 Then      ' Computer Skills
    If cboYesNo(27).Text = "Yes" Then chkRecommendation(8).Value = 1
ElseIf Index = 28 Then
    If cboActivity.ListIndex = 2 Or cboActivity.ListIndex = 3 Then cboYesNo(Index).ListIndex = -
End If
End Sub

Private Sub cboYesNo_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

Select Case KeyCode
    Case vbKeyDown
        IsArrowKey = KeyCode
    Case vbKeyRight
        IsArrowKey = KeyCode
    Case vbKeyLeft
        IsArrowKey = KeyCode
    Case vbKeyRight
        IsArrowKey = KeyCode
End Select
End Sub

Private Sub cboYesNo_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If KeyCode = vbKeyBack Then
    cboYesNo(Index).ListIndex = -1
End If
End Sub

Private Sub cboYesNo_LostFocus(Index As Integer)
Dim i As Integer

If Index = 0 Then
    If CheckForTab Then
        If cboYesNo(0).Text = "Yes" Then txtTriage(1).SetFocus
    End If
ElseIf Index = 1 Then
    If CheckForTab Then
        If cboYesNo(1).Text = "Yes" Then txtTriage(1).SetFocus
    End If
ElseIf Index = 2 Then
    If CheckForTab Then
        If cboYesNo(2).Text = "No" Then
            cboYesNo(3).TabStop = False
            cboYesNo(4).TabStop = False
            txtTriage(2).TabStop = False
        Else
            cboYesNo(3).TabStop = True
            cboYesNo(4).TabStop = True
            txtTriage(2).TabStop = True
        End If
    End If
ElseIf Index = 4 Then
    If cboYesNo(4).Text = "Yes" And cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value =

    If CheckForTab Then tabTriage.Tab = 1
ElseIf Index = 5 Then
    If CheckForTab Then
        If cboYesNo(5).Text = "Yes" Then
            cboYesNo(6).SetFocus
        Else
            cboOverallHealth(1).SetFocus
        End If
    End If
ElseIf Index = 6 Then
    If CheckForTab Then
        If cboYesNo(6).Text = "Yes" Then
            chkRecommendation(2).Value = 1
        End If
    End If
End Sub
```



```
        chkRecommendation(3).Value = 1
    End If
End If
ElseIf Index = 9 Then
    If CheckForShftTab Then
        If cboOverallHealth(1).ListIndex = 0 Or cboOverallHealth(1).ListIndex = 1 Then
            cboOverallHealth(1).SetFocus
        End If
    ElseIf CheckForTab Then
        If cboYesNo(9).Text = "Yes" Then
            lstHealthCondition.SetFocus
        Else
            txtTriage(4).SetFocus
        End If
    End If
ElseIf Index = 10 Then
    If CheckForTab Then
        If cboYesNo(Index).Text = "Yes" Then
            If cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value = 1
            cmdMovement.SetFocus
        End If
    End If
ElseIf Index = 11 Then
    If CheckForTab Then
        If cboYesNo(10).Text = "Yes" Then
            cboYesNo(11).ListIndex = -1
            cmdMovement.SetFocus
        ElseIf cboYesNo(11).Text = "Yes" Then
            If cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value = 1
        End If
    End If
ElseIf Index = 12 Then
    If CheckForShftTab Then
        If IsSighted Or cboYesNo(2).Text = "No" Then
            cmdProblem.SetFocus
        Else
            cboYesNo(11).SetFocus
        End If
    ElseIf CheckForTab Then
        If cboYesNo(Index).Text = "Yes" Then
            If cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value = 1
            chkRecommendation(6).Value = 1
            cmdHousehold.SetFocus
        End If
    End If
ElseIf Index = 13 Then
    If CheckForTab Then
        If cboYesNo(12).ListIndex = 0 Then
            cboYesNo(13).ListIndex = -1
            cmdHousehold.SetFocus
        ElseIf cboYesNo(Index).Text = "Yes" Then
            If cboYesNo(2).ListIndex = 0 Then chkRecommendation(1).Value = 1
            chkRecommendation(6).Value = 1
        End If
    End If
ElseIf Index = 14 Then
    If CheckForTab Then
        For i = 0 To 2
            'If partially sighted, rec LV
            If cboYesNo(i).Text = "Yes" Then chkRecommendation(1).Value = 1
        Next

        If cboYesNo(Index).Text = "Yes" Then
            chkRecommendation(5).Value = 1
            tabTriage.Tab = 3
            cmdPsychological.SetFocus
        End If
    End If
ElseIf Index = 15 Then
    If CheckForShftTab Then
        cboYesNo(14).SetFocus
    End If
End If
```

```
ElseIf CheckForTab Then
    If cboYesNo(14).ListIndex = 0 Then
        cboYesNo(15).ListIndex = -1
        tabTriage.Tab = 3
        cmdPsychological.SetFocus
    ElseIf cboYesNo(15).ListIndex = 0 Then
        chkRecommendation(5).Value = 1
        tabTriage.Tab = 3
        cmdPsychological.SetFocus
    Else
        tabTriage.Tab = 3
        cboYesNo(16).SetFocus
    End If
End If
ElseIf Index = 16 Then
    If CheckForShiftTab Then
        IsShiftTab = True
        tabTriage.Tab = 2
        cboYesNo(15).SetFocus
    ElseIf CheckForTab Then
        If cboYesNo(15).ListIndex = 0 Then
            cboYesNo(16).ListIndex = -1
            cmdPsychological.SetFocus
        ElseIf cboYesNo(16).ListIndex = 0 Then
            chkRecommendation(5).Value = 1
            tabTriage.Tab = 3
            cmdPsychological.SetFocus
        End If
    End If
ElseIf Index = 17 Then
    If CheckForTab Then
        If cboYesNo(16).ListIndex = 0 Then
            cboYesNo(17).ListIndex = -1
            cmdPsychological.SetFocus
        ElseIf cboYesNo(Index).ListIndex = 0 Then
            chkRecommendation(5).Value = 1
            tabTriage.Tab = 3
            cmdPsychological.SetFocus
        End If
    End If
ElseIf Index = 18 Or Index = 19 Then
    If CheckForTab Then
        If Index = 19 And Trim(txtTriage(20)) <> "" Then
            If CInt(txtTriage(20)) < 18 Then
                tabTriage.Tab = 4
                cboActivity.SetFocus
            End If
        End If
        If cboYesNo(Index).Text = "Yes" Then chkRecommendation(4).Value = 1
    End If
    If Index = 19 Then
        If cboYesNo(19).ListIndex <> -1 And cboYesNo(18).ListIndex = -1 Then
            MsgBox "Please answer Question 1 !", vbInformation + vbOKOnly, "Psychological Status"
            cboYesNo(18).SetFocus
        ElseIf cboYesNo(19).ListIndex = -1 And cboYesNo(18).ListIndex <> -1 Then
            MsgBox "Please answer Question 2 !", vbInformation + vbOKOnly, "Psychological Status"
            cboYesNo(19).SetFocus
        End If
    End If
ElseIf Index = 20 Then
    If CheckForTab Then
        If cboYesNo(Index).Text = "No" Then cboYesNo(23).SetFocus
    End If
ElseIf Index = 21 Then
    If cboYesNo(Index).Text = "Yes" Then chkRecommendation(9).Value = 1
ElseIf Index = 22 Then
```

```

If Not CheckForShiftTab Then
    If cboYesNo(Index).Text = "Yes" Then
        chkRecommendation(9).Value = 1
        cboActionStatus.ListIndex = 1
    End If
    cboYesNo(25).SetFocus
End If
ElseIf Index = 23 Then
    If CheckForTab Then
        If cboYesNo(Index).Text = "Yes" Then
            If cboAgree(0).Text = "Yes" Then 'Yes to like to be employed and is Citizen
                chkRecommendation(9).Value = 1
                cboYesNo(25).SetFocus
            ElseIf cboAgree(0).Text = "No" Then 'Yes to like be employed and is not citizen
                cboYesNo(24).SetFocus
            End If
        ElseIf cboYesNo(Index).Text = "No" Then
            If Trim(txtTriage(20)) <> "" Then
                If CInt(txtTriage(20)) >= 18 Or CInt(txtTriage(20)) <= 64 Then chkRecommendation(9).
                value = 1
            End If
        End If
    End If
ElseIf Index = 24 Then
    If CheckForTab Then
        If cboYesNo(Index).Text = "Yes" Then
            chkRecommendation(9).Value = 1
        ElseIf cboYesNo(Index).Text = "No" Then
            cboYesNo(26).SetFocus
        End If
    End If
ElseIf Index = 25 Then
    If cboYesNo(Index).Text = "Yes" Then chkRecommendation(10).Value = 1
ElseIf Index = 26 Then
    If CheckForTab Then
        tabTriage.Tab = 4
        If cboYesNo(26).Text = "Yes" Then chkRecommendation(8).Value = 1 ' Computer Skills
    End If
ElseIf Index = 27 Then
    If CheckForShiftTab Then
        IsShiftTab = True
        tabTriage.Tab = 3
        cboYesNo(26).SetFocus
        If cboYesNo(27).Text = "Yes" Then chkRecommendation(8).Value = 1 ' Computer Skills
    End If
ElseIf Index = 28 Then
    If cboYesNo(Index).Text = "Yes" Then chkRecommendation(0).Value = 1
ElseIf Index = 30 Then
    If CheckForTab Then
        If cboYesNo(30).Text <> "Yes" Then txtTriage(15).SetFocus
    End If
End If

If IsArrowKey <> 0 Then IsArrowKey = 0
End Sub

Private Sub chkFactors_LostFocus(Index As Integer)
    If chkFactors(Index).Value = 1 Then chkRecommendation(0).Value = 1
    If Index = 12 And CheckForTab Then tabTriage.Tab = 5
End Sub

Private Sub cmdAppointments_Click()
    MsgBox "Make assessment appointments in each accepted area. Instruct consumer to bring" & _
        " any eyeglasses or optical devices to appointments", vbInformation + vbOKOnly, "Appoime
s"
End Sub

Private Sub cmdAppointments_KeyUp(KeyCode As Integer, Shift As Integer)

```

```
If KeyCode = vbKeyBack And (Shift And vbShiftMask) > 0 Then tabTriage.Tab = 7
End Sub

Private Sub cmdAppointments_LostFocus()
    If CheckForShiftTab Then
        IsShiftTab = True
        cboActionStatus.SetFocus
        tabTriage.Tab = 6
    End If
End Sub

Private Sub cmdAppointments_Click()
    MsgBox "Make assessment appointments in each accepted area. Instruct consumer to bring any eyes  
asses or optical devices to appointments", vbOKOnly, "Appointment"
    tabTriage.Tab = 0
    cboSex.SetFocus
    tabTriage.Tab = 6
    txtTriage(16).SetFocus
End Sub

Private Sub cmdAppointments_LostFocus()
    If CheckForShiftTab Then
        tabTriage.Tab = 6
        cboActionStatus.SetFocus
    End If
End Sub

Private Sub cmdClose_Click()
    Unload Me
End Sub

Private Sub cmdFactors_Click()
    Dim response As Integer

    tabTriage.Tab = 3
    response = MsgBox("Recommendations for Social Service Assessment - check all that apply", vbOKO  
y, "Risk Factors")
    If response = vbOK Then
        tabTriage.Tab = 4
        chkFactors(0).SetFocus
    End If
End Sub

Private Sub cmdFactors_LostFocus()
    If CheckForShiftTab Then
        If cboFrequency2.ListIndex = 0 Or cboFrequency2.ListIndex = 1 Then
            cboYesNo(29).SetFocus
        Else
            cboFrequency2.SetFocus
        End If
    End If
End Sub

Private Sub cmdInstruction_Click()
    MsgBox "If ANY recommendations are accepted, EYE REPORT must be requested.", vbOKOnly, "Service  
nstruction"
    tabTriage.Tab = 0
    cboSex.SetFocus
    tabTriage.Tab = 6
    txtTriage(11).SetFocus
End Sub

Private Sub cmdInstruction_LostFocus()
    If CheckForShiftTab Then
        IsShiftTab = True
        tabTriage.Tab = 5
        txtTriage(10).SetFocus
    End If
End Sub
```

End Sub

Private Sub cmdClear\_Click()

IsNewRecord = True  
IsFileDateChecked = False  
clearData cboStaffs, False  
clearData cboYesNo, True  
clearData txtTriage, True  
clearData lstHealthCondition, False  
clearData cboAcceptReject, True  
clearData cboRejectionReason, True  
clearData cboActionStatus, False  
clearData cboActivity, False  
clearData cboFrequency2, False  
clearData chkFactors, True  
clearData chkRecommendation, True  
clearData cboOverallHealth, True  
clearData cboEpisodeOutcome, False  
clearData cboDenyReason, False  
clearData cbo7OB, True  
clearData chk7OB, True

With rstTriage

    If Not .BOF Then

        .MoveFirst  
        LoadDataToControls rstTriage, cboSex, False  
        cboSex.Locked = True  
        txtTriage(0).Text = .Fields.Item("Birthdate")  
        txtTriage(0).Locked = True  
        LoadDataToControls rstTriage, cboAgree, True  
        LoadDataToControls rstTriage, cboAgree, True  
        LoadDataToControls rstTriage, lstRace, False

    End If

End With

cmdSave.Enabled = True

tabTriage.Tab = 0

cboSex.SetFocus

End Sub

Private Sub cmdNext\_Click()

If tabTriage.Tab = 7 Then

    MsgBox "It is last page for Triage already!", vbInformation + vbOKOnly, "Next Page"

Else

    tabTriage.Tab = tabTriage.Tab + 1

    frmTriage.Caption = getTabCaption(tabTriage.Tab)

End If

End Sub

Private Sub cmdNExtRecord\_Click()

With rstTriage

    .MoveNext

    If .AbsolutePosition = 1 Then

        cmdPreviousRecord.Enabled = False

        cmdNextRecord.Enabled = True

    ElseIf .AbsolutePosition < .RecordCount Then

        cmdPreviousRecord.Enabled = True

        cmdNextRecord.Enabled = True

    ElseIf .RecordCount = .AbsolutePosition Then

        cmdPreviousRecord.Enabled = True

        cmdNextRecord.Enabled = False

    End If

txtEpisode = !Episode

LoadDataToControls rstTriage, cboStaffs, False

LoadDataToControls rstTriage, cboSex, False

LoadDataToControls rstTriage, cboYesNo, True

```

LoadDataToControls rstTriage, txtTriage, True
LoadDataToControls rstTriage, lstHealthCondition, False
LoadDataToControls rstTriage, cboAcceptReject, True
LoadDataToControls rstTriage, cboRejectionReason, True
LoadDataToControls rstTriage, cboActionStatus, False
LoadDataToControls rstTriage, cboActivity, False
LoadDataToControls rstTriage, cboFrequency2, False
LoadDataToControls rstTriage, chkFactors, True
LoadDataToControls rstTriage, cbo7OB, True
LoadDataToControls rstTriage, cboServiceLocation, False
LoadDataToControls rstTriage, cboEpisodeOutcome, False
LoadDataToControls rstTriage, cboDenyReason, False
LoadDataToControls rstTriage, chk7OB, True

If ![FinishDate] = "12:00:00 PM" Then
    cmdSave.Enabled = True
Else
    cmdSave.Enabled = False
End If

If IsDate(![FinishDate] & vbNullString) And ![FinishDate] & vbNullString <> "12:00:00 PM"
    IsFileDateChecked = True
End If
End With

tabTriage.Tab = 0
End Sub

Private Sub cmdPrevious_Click()

If tabTriage.Tab = 0 Then
    MsgBox "It is first page for Triage already!", vbInformation + vbOKOnly, "Previous Page"
Else
    tabTriage.Tab = tabTriage.Tab - 1
    frmTriage.Caption = getTabCaption(tabTriage.Tab)
End If
End Sub

Private Sub cmdPreviousEpisode_Click()

With rstTriage
    .MovePrevious
    If .AbsolutePosition = 1 Then
orm
        cmdPreviousRecord.Enabled = False
        cmdNextRecord.Enabled = True
    Else
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = True
    End If

    If ![FinishDate] = "12:00:00 PM" Then
        cmdSave.Enabled = True
    Else
        cmdSave.Enabled = False
    End If

    txtEpisode = !Episode
    LoadDataToControls rstTriage, cboYesNo, True
    LoadDataToControls rstTriage, txtTriage, True
    LoadDataToControls rstTriage, lstHealthCondition, False
    LoadDataToControls rstTriage, cboAcceptReject, True
    LoadDataToControls rstTriage, cboRejectionReason, True
    LoadDataToControls rstTriage, cboActionStatus, False
    LoadDataToControls rstTriage, cboActivity, False
    LoadDataToControls rstTriage, cboFrequency2, False
    LoadDataToControls rstTriage, chkFactors, True
End With

tabTriage.Tab = 0

```

```
txtEpisode.SetFocus
End Sub
```

```
Private Sub cmdPreviousRecord_Click()
```

```
With rstTriage
```

```
    .MovePrevious
```

```
    If .AbsolutePosition = 1 Then
        cmdPreviousRecord.Enabled = False
        cmdNextRecord.Enabled = True
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = False
    End If
```

```
txtEpisode = !Episode
```

```
LoadDataToControls rstTriage, cboStaffs, False
LoadDataToControls rstTriage, cboSex, False
LoadDataToControls rstTriage, cboYesNo, True
LoadDataToControls rstTriage, txtTriage, True
LoadDataToControls rstTriage, lstHealthCondition, False
LoadDataToControls rstTriage, cboAcceptReject, True
LoadDataToControls rstTriage, cboRejectionReason, True
LoadDataToControls rstTriage, cboActionStatus, False
LoadDataToControls rstTriage, cboActivity, False
LoadDataToControls rstTriage, cboFrequency2, False
LoadDataToControls rstTriage, chkFactors, True
LoadDataToControls rstTriage, cbo7OB, True
LoadDataToControls rstTriage, cboServiceLocation, False
LoadDataToControls rstTriage, cboEpisodeOutcome, False
LoadDataToControls rstTriage, cboDenyReason, False
LoadDataToControls rstTriage, chk7OB, True
```

```
If ![FinishDate] = "12:00:00 PM" Then
    cmdSave.Enabled = True
Else
    cmdSave.Enabled = False
End If
```

```
If IsDate(![FinishDate] & vbNullString) And ![FinishDate] & vbNullString <> "12:00:00 PM"
    IsFileDateChecked = True
End If
```

```
End With
```

```
tabTriage.Tab = 0
End Sub
```

```
Private Sub cmdProblem_Click()
Dim response As Integer
```

```
On Error GoTo Errhandler
```

```
tabTriage.Tab = 3
response = MsgBox("As a result of your vision impairment, are you having problems in the following areas?" & _
    " If you receive a yes answer to any questions move on to the next area unless otherwise specified." & _
    " A yes prompts a recommendation for assessment in that area.", vbOKOnly, "Problem")
If response = vbOK Then
    tabTriage.Tab = 2
End If
Exit Sub
```

```
Errhandler:
MsgBox Err.Description & " " & Err.Number, vbCritical + vbOKOnly, "Problem Click"
End Sub
```

```
Private Sub cmdProblem_LostFocus()
```

```
On Error GoTo Errhandler
```

```
If CheckForShiftTab Then
```

```
IsShiftTab = True
```

```
tabTriage.Tab = 1
```

```
txtTriage(9).SetFocus
```

```
ElseIf CheckForTab And (IsSighted Or cboYesNo(2).Text = "No") Then
```

```
cmdMovement.SetFocus
```

```
End If
```

```
Exit Sub
```

```
Errhandler:
```

```
MsgBox Err.Description & " " & Err.Number, vbCritical + vbOKOnly, "Problem Lost Focus"
```

```
End Sub
```

```
Private Sub cmdRecommendation_Click()
```

```
Dim Str As String
```

```
Dim response As Integer
```

```
Str = "Describe training and services that could address the problems identified. Recommend additional services if necessary." & vbNewLine & vbNewLine
```

```
Str = Str & "Recommendations Rules:" & vbNewLine
```

```
Str = Str & " Age 18-64: Career Services" & vbNewLine
```

```
Str = Str & " Age 14-64 and agree to career services: Career CP. If 14-64 and rejects career services, SWCP (notify CP to suggest Career Services)" & vbNewLine
```

```
Str = Str & " Assign a Career CP On-Call if career services are desired (academic, Computer skills) and status is Ordinary" & vbNewLine
```

```
Str = Str & " If employed and agrees to LV (but not Career): SWCP and recommend Career Assessment" & vbNewLine
```

```
Str = Str & " Assign a SWCP and recommend Social Service Assessment if any Risk Factors were indicated, and also if caller is age 80 or older" & vbNewLine
```

```
Str = Str & " All cases marked as Crises or Urgent: SWCP - Intensive level, UNLESS the only factor is danger or losing job, then assign Career CP - Intensive level" & vbNewLine
```

```
Str = Str & " All cases that do not meet the above criteria - SWCP On-Call" & vbNewLine
```

```
Str = Str & vbNewLine
```

```
MsgBox Str, vbOKOnly, "Recommendation"
```

```
Str = "Based on your responses, the following services would be beneficial to you."
```

```
Str = Str & " Mark each recommended area with Accepted Or Rejected, and if rejected, a reason"
```

```
MsgBox Str, vbOKOnly, "Recommendation"
```

```
tabTriage.Tab = 0
```

```
cboSex.SetFocus
```

```
tabTriage.Tab = 5
```

```
chkRecommendation(0).SetFocus
```

```
End Sub
```

```
Private Sub cmdRecommendation_LostFocus()
```

```
Dim i As Integer
```

```
Dim AcceptServices As Boolean
```

```
If CheckForShiftTab Then
```

```
IsShiftTab = True
```

```
tabTriage.Tab = 5
```

```
ElseIf CheckForTab Then
```

```
For i = 0 To chkRecommendation.Count - 1
```

```
If chkRecommendation(i).Value = 1 Then
```

```
ServiceLocation = i
```

```
If tabTriage.Tab = 5 Then chkRecommendation(i).SetFocus
```

```
Exit Sub
```

```
End If
```

```
Next
```

```
If ServiceLocation = 0 Then
```

```
tabTriage.Tab = 7
```

```
cmdInstruction.SetFocus
```

```
End If
```

```
End If
```



End Sub

```
Private Sub cmdSave_Click()
Dim i As Integer
Dim tmpStr As String
Dim colTriage As New Collection
Dim rst As New ADODB.Recordset
```

On Error GoTo errSave

```
If Not IsFileDateChecked Then
txtTriage(19) = "" 'Guarantee No End Date, then No episode close date
If IsNewRecord Then txtEpisode = frmMainMenu.cboEpisode.ListCount + 1
If Trim(txtRecordNo) = "0" Then txtEpisode = 1.
colTriage.Add cboStaffs
colTriage.Add cboSex
colTriage.Add cboAgree
colTriage.Add lstRace
colTriage.Add cboYesNo
colTriage.Add txtTriage
colTriage.Add txtEpisode
colTriage.Add lstHealthCondition
colTriage.Add cboAcceptReject
colTriage.Add cboRejectionReason
colTriage.Add cboActionStatus
colTriage.Add cboActivity
colTriage.Add cboFrequency2
colTriage.Add chkFactors
colTriage.Add chkRecommendation
colTriage.Add cboOverallHealth
colTriage.Add cbo7OB
colTriage.Add chk7OB
If SaveDataToDB(frmTriage, rstTriage, colTriage, "0", False) Then 'Call public func
on to save data to DB
MsgBox "Successfully saved the Triage information!", vbOKOnly, "Save"
```

```
If IsDate(Trim(txtTriage(18))) Then
For i = 0 To chkRecommendation.Count - 1
If chkRecommendation(i).Value = 1 And cboAcceptReject(i).Text = "Accepted" Then
If i = 0 Then
tmpStr = "010"
ElseIf i = 1 Then
tmpStr = "041"
ElseIf i = 2 Then
tmpStr = "047"
ElseIf i = 3 Then
tmpStr = "045"
ElseIf i = 4 Then
tmpStr = "011"
ElseIf i = 5 Then
tmpStr = "013"
ElseIf i = 6 Then
tmpStr = "014"
ElseIf i = 7 Then
tmpStr = "031"
ElseIf i = 8 Then
tmpStr = "160"
ElseIf i = 9 Then
tmpStr = "050"
ElseIf i = 10 Then
tmpStr = "056"
ElseIf i = 11 Then
tmpStr = "141"
ElseIf i = 12 Then
tmpStr = "121"
End If
```

```
If tmpStr <> "" Then
cnn.Execute "Insert into Interventions(ConsumerID,Agency,Episode,CaseNumber,
erventionID,StartDate) " & _
```

```

        "Values(" & CustomerID & ", " & AgencyCode & ", " & Trim(txtEpisode) & ", " &
aseNumber & ", " & tmpStr & _
        " " & Date & " " & " ) "
    End If
    tmpStr = ""

    End If
Next
End If
cmdSave.Enabled = True
If IsDate(Trim(txtTriage(18))) And Trim(txtTriage(18)) <> "12:00:00 PM" Then IsFileDateCh
ked = True
ElseIf IsDate(Trim(txtTriage(18))) And IsDate(Trim(txtTriage(19))) Then
    cmdSave.Enabled = False
End If

ElseIf IsDate(Trim(txtTriage(19))) And Trim(txtTriage(19)) <> "12:00:00 PM" Then ' Update Epis
eCloseDate after all the work is done
    cnn.Execute "Update Triage Set EpisodeCloseDate = #" & txtTriage(19) & "# " & _
        "Where ConsumerID = " & CustomerID & " And Agency = " & AgencyCode & " " & _
        "And Episode = " & Episode & " And CaseNumber = " & CaseNumber & " "
End If

IsNewRecord = False
Set colTriage = Nothing
Exit Sub

errSave:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save"
End Sub

Private Sub cmdShortKeys_Click()
MsgBox "Press Alt O for shortcuts" & vbNewLine & "Press Alt I to Save Payer Information" & vbNe
ine & _
"Press Alt A for the Authorizations Screen" & vbNewLine & "Press Alt W to enter a New Payer" &
NewLine & _
"Press Alt E to move to the Previous Payer" & vbNewLine & "Press Alt F to move to the Next Paye
, vbOKOnly, ""
End Sub

Private Sub Form_Load()

Set rstTriage = New ADODB.Recordset
rstTriage.CursorLocation = adUseClient
End Sub

Private Sub Form_Unload(Cancel As Integer)

frmMainMenu.cboEpisode.Clear
rstTriage.Requery
With rstTriage
    If Not (.BOF And .EOF) Then
        .MoveFirst
        While Not .EOF
            frmMainMenu.cboEpisode.AddItem ![[Episode]]
            .MoveNext
        Wend
    End If
End With

frmMainMenu.txtEpisodeCloseDate = txtTriage(19)
If txtTriage(19) = "" Then
    frmMainMenu.Label2.Visible = False
    frmMainMenu.txtEpisodeCloseDate.Visible = False
Else
    frmMainMenu.Label2.Visible = True
    frmMainMenu.txtEpisodeCloseDate.Visible = True
End If

```

```

If rstTriage.State = adStateOpen Then rstTriage.Close
Set rstTriage = Nothing
Set frmTriage = Nothing
End Sub

```

```

Private Sub lstHealthCondition_KeyUp(KeyCode As Integer, Shift As Integer)
Dim i As Integer

```

```

If KeyCode = vbKeyBack Then
If lstRace.SelCount = 1 Then
lstHealthCondition.Selected(lstHealthCondition.ListIndex) = False
ElseIf lstHealthCondition.ListCount > 1 Then
For i = lstHealthCondition.ListCount - 1 To 0 Step -1
If lstHealthCondition.Selected(i) Then
lstHealthCondition.Selected(i) = False
End If
Next
End If
End If
End Sub

```

```

Private Sub lstHealthCondition_LostFocus()

```

```

If CheckForTab And lstHealthCondition.Selected(28) Then
txtTriage(3).SetFocus
ElseIf CheckForTab And Not lstHealthCondition.Selected(28) Then
txtTriage(4).SetFocus
End If
End Sub

```

```

Private Sub Form_Activate()
Dim i As Integer, j As Integer, k As Integer
Dim today As Date

```

```

On Error GoTo Errhandler

```

```

txtFirstName = FirstName
txtLastName = LastName
txtCaseNumber = CaseNumber
LoadControlsItems cboServiceLocation, False
LoadControlsItems cboStaffs, False

```

```

rstTriage.Open "Select Trouble From Registration Where ConsumerID = " & CustomerID & " And Agen
= ' ' & _

```

```

AgencyCode & " ' And CaseNumber = ' ' & CaseNumber & " ' ", cnn, adOpenForwardOnly,

```

```

LockReadOnly

```

```

With rstTriage

```

```

If Not (.BOF And .EOF) Then

```

```

If ![Trouble] = 2 Then

```

```

IsSighted = True

```

```

Else

```

```

IsSighted = False

```

```

End If

```

```

End If

```

```

End With

```

```

rstTriage.Close

```

```

rstTriage.Open "Select * From Triage Where ConsumerID = " & CustomerID & " And Agency = ' ' & _
AgencyCode & " ' And CaseNumber = ' ' & CaseNumber & " ' Order By TriageSqe", cnn, _
adOpenDynamic, adLockOptimistic

```

```

With rstTriage

```

```

If Not (.BOF And .EOF) Then

```

```

rstTriage.AbsolutePosition = frmMainMenu.cboEpisode.Text

```

```

txtRecordNo = .RecordCount

```

```

txtEpisode = frmMainMenu.cboEpisode.Text

```

```

LoadDataToControls rstTriage, cboStaffs, False

```

```

LoadDataToControls rstTriage, txtTriage, True

```

```

LoadDataToControls rstTriage, cboAgree, True

```

```

LoadDataToControls rstTriage, cboSex, False
LoadDataToControls rstTriage, lstRace, False
LoadDataToControls rstTriage, cboYesNo, True
LoadDataToControls rstTriage, lstHealthCondition, False
LoadDataToControls rstTriage, cboAcceptReject, True
LoadDataToControls rstTriage, cboRejectionReason, True
LoadDataToControls rstTriage, cboActionStatus, False
LoadDataToControls rstTriage, cboActivity, False
LoadDataToControls rstTriage, cboFrequency2, False
LoadDataToControls rstTriage, chkFactors, True
LoadDataToControls rstTriage, chkRecommendation, True
LoadDataToControls rstTriage, cboOverallHealth, True
LoadDataToControls rstTriage, cbo7OB, True
LoadDataToControls rstTriage, cboServiceLocation, False
LoadDataToControls rstTriage, cboEpisodeOutcome, False
LoadDataToControls rstTriage, cboDenyReason, False
LoadDataToControls rstTriage, chk7OB, True

```

```

If .RecordCount > 1 Then
    If .AbsolutePosition = 1 Then
        cmdPreviousRecord.Enabled = False
        cmdNextRecord.Enabled = True
    ElseIf .AbsolutePosition < .RecordCount Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = True
    ElseIf .RecordCount = .AbsolutePosition Then
        cmdPreviousRecord.Enabled = True
        cmdNextRecord.Enabled = False
    End If
Else
    cmdNextRecord.Enabled = False
    cmdPreviousRecord.Enabled = False
End If

```

```

If IsDate(![FinishDate] & vbNullString) And ![FinishDate] & vbNullString <> "12:00:00 PM"
hen
    IsFileDateChecked = True
End If
End If
End With

```

```

' If Episode is closed then disable the Payer Save and Triage Save buttons

```

```

If IsDate(txtTriage(19)) Then
    cmdSave.Enabled = False
End If

```

```

tabTriage.Tab = 0
txtFirstName.SetFocus
Exit Sub

```

```

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Form Activate"
End Sub

```

```

Private Sub lstRace_KeyUp(KeyCode As Integer, Shift As Integer)
Dim i As Integer

```

```

If KeyCode = vbKeyBack Then
    If lstRace.SelCount = 1 Then
        lstRace.Selected(lstRace.ListIndex) = False
    ElseIf lstRace.ListCount > 1 Then
        For i = lstRace.ListCount - 1 To 0 Step -1
            If lstRace.Selected(i) Then
                lstRace.Selected(i) = False
            End If
        Next
    End If
End If
End If

```

End Sub

Private Sub lstRace\_LostFocus()

If CheckForTab Then

    If IsSighted Then

        tabTriage.Tab = 1

        cboOverallHealth(0).SetFocus

    End If

End If

End Sub

Private Sub tabTriage\_Click(PreviousTab As Integer)

Dim IsAccepted As Boolean

Dim i As Integer

On Err GoTo Errhandler

If PreviousTab = 0 Then

    If txtTriage(0) = "" Then

        MsgBox "Please enter Birthday in this field !", vbInformation + vbOKOnly, "Birthday"

        tabTriage.Tab = 0

        txtTriage(0).SetFocus

    Exit Sub

    End If

    If cboAgree(0) = "" Then

        MsgBox "Please answer resident status !", vbInformation + vbOKOnly, "Resident Status"

        tabTriage.Tab = 0

        cboAgree(0).SetFocus

    Exit Sub

    End If

End If

If tabTriage.Tab = 0 Then

    cboSex.SetFocus

    frmTriage.Caption = "A. Demographics"

ElseIf tabTriage.Tab = 1 Then

    If Not IsShftTab Then cboOverallHealth(0).SetFocus

    IsShftTab = False

    frmTriage.Caption = "B. Health"

ElseIf tabTriage.Tab = 2 Then

    If Not IsShftTab Then cmdProblem.SetFocus

    IsShftTab = False

    frmTriage.Caption = "C. Problem ID"

ElseIf tabTriage.Tab = 3 Then

    If Not IsShftTab Then cboYesNo(16).SetFocus

    IsShftTab = False

    frmTriage.Caption = "D. Problem ID"

ElseIf tabTriage.Tab = 4 Then

    If Not IsShftTab Then cboYesNo(27).SetFocus

    IsShftTab = False

    If cboYesNo(26).ListIndex = 0 Then cmdSocialNeeds.SetFocus

    frmTriage.Caption = "E. Risk Factors"

ElseIf tabTriage.Tab = 5 Then

    If Not IsShftTab Then cmdRecommendation.SetFocus

    IsShftTab = False

    frmTriage.Caption = "F. Recommendations"

ElseIf tabTriage.Tab = 6 Then

    If Not IsShftTab Then cmdInstruction.SetFocus

    IsShftTab = False

    frmTriage.Caption = "G. Eye Report"

ElseIf tabTriage.Tab = 7 Then

    If Not IsShftTab Then cbo7OB(0).SetFocus

    frmTriage.Caption = "7-OB"

    IsShftTab = False

For i = 0 To chkRecommendation.Count - 1

    If chkRecommendation(i).Value = 1 Then

```

    If cboAcceptReject(i).Text = "Accepted" Then IsAccepted = True
End If
Next
End If
Exit Sub

Errhandler:
MsgBox Err.Description & " " & Err.Number, vbCritical + vbOKOnly, "Tab Click"
End Sub

```

```

Private Function Check_Close_Episode() ' Make sure all goals, objectives, programs are fir
shed ' before we can close the episode
Dim rstCheck As ADODB.Recordset
Dim tmpBuf(10) As String
Dim tmpMessage As String
Dim w As Integer
Dim v As Integer

On Error GoTo errSave:

Set rstCheck = New ADODB.Recordset

Check_Close_Episode = False

' All objectives must be finished

w = 0
v = 0

rstCheck.Open "Select Distinct Description From ObjectiveAchieve OA, InterventionCodes IC Where
A.ConsumerID = " & CustomerID & _
" And OA.Agency = " & AgencyCode & " And OA.CaseNumber = " & CaseNumber & _
" And OA.Episode = " & Episode & " And (Achievement IS NULL " & _
" Or Trim(Achievement) = " & Chr(39) & Chr(39) & " Or CompletionDate IS NULL Or
ompletionDate = #12:00:00 PM#" & _
" And OA.InterventionID = IC.Code", cnn, adOpenForwardOnly, adLockReadOnly

If rstCheck.RecordCount > 0 Then
    With rstCheck
        While Not (.EOF)
            If w > 0 Then
                tmpBuf(v) = tmpBuf(v) & " and "
            End If
            tmpBuf(v) = tmpBuf(v) & ![Description]
            .MoveNext
            w = w + 1
        Wend
        End With
        tmpBuf(v) = "There are " & tmpBuf(v) & " Objectives that are not completed"
        Check_Close_Episode = True
    End If

    rstCheck.Close

    ' All goals must be finished

    w = 0
    v = v + 1

    rstCheck.Open "Select Distinct Description From GoalAchieve GA, InterventionCodes IC Where GA.C
sumerID = " & CustomerID & _
" And GA.Agency = " & AgencyCode & " And GA.CaseNumber = " & CaseNumber & _
" And GA.Episode = " & Episode & " And (Achievement IS NULL " & _
" Or Trim(Achievement) = " & Chr(39) & Chr(39) & " Or CompletionDate IS NULL Or
ompletionDate = #12:00:00 PM#" & _
" And GA.InterventionID = IC.Code", cnn, adOpenForwardOnly, adLockReadOnly

    If rstCheck.RecordCount > 0 Then

```

```

With rstCheck
  While Not (.EOF)
    If w > 0 Then
      tmpBuf(v) = tmpBuf(v) & " and "
    End If
    tmpBuf(v) = tmpBuf(v) & ![Description]
    .MoveNext
    w = w + 1
  Wend
End With
tmpBuf(v) = vbNewLine & "There are " & tmpBuf(v) & " Goals that are not completed"
Check_Close_Episode = True
End If

rstCheck.Close

' All progress reports must be finished

w = 0
v = v + 1

rstCheck.Open "Select Distinct Description From ProgressReport PR, InterventionCodes IC " & _
  "Where PR.ConsumerID = " & CustomerID & _
  " And PR.Agency = " & AgencyCode & " And PR.CaseNumber = " & CaseNumber & _
  " And PR.Episode = " & Episode & " And (StaffName IS NULL " & _
  " Or Trim(StaffName) = " & Chr(39) & Chr(39) & " Or Summary IS NULL Or Trim(Sum
ry) = " & Chr(39) & Chr(39) & _
  " Or PeriodFrom IS NULL Or PeriodFrom = #12:00:00 PM# Or PeriodTo IS NULL Or Pe
odTo = #12:00:00 PM#" & _
  " And PR.InterventionID = IC.Code"

If rstCheck.RecordCount > 0 Then
  With rstCheck
    While Not (.EOF)
      If w > 0 Then
        tmpBuf(v) = tmpBuf(v) & " and "
      End If
      tmpBuf(v) = tmpBuf(v) & ![Description]
      .MoveNext
      w = w + 1
    Wend
  End With
  tmpBuf(v) = vbNewLine & "There are " & tmpBuf(v) & " Progress Reports that are not completed"
  Check_Close_Episode = True
End If

rstCheck.Close

' All Assessments must be finished

w = 0
v = v + 1

rstCheck.Open "Select Distinct Description From MobilityAssessments A, InterventionCodes IC " & _
  "Where A.ConsumerID = " & CustomerID & _
  " And A.Agency = " & AgencyCode & " And A.CaseNumber = " & CaseNumber & _
  " And A.Episode = " & Episode & _
  " And (FinishDate IS NULL Or FinishDate = #12:00:00 PM#" & _
  " And A.InterventionID = IC.Code", cnn, adOpenForwardOnly, adLockReadOnly

If rstCheck.RecordCount > 0 Then
  tmpBuf(v) = vbNewLine & "There is a Mobility Assessment that is not complete"
  Check_Close_Episode = True
End If

rstCheck.Close

```

```
w = 0
v = v + 1
```

```
rstCheck.Open "Select Distinct Description From ComputerAssessment A, InterventionCodes IC " & _
    "Where A.ConsumerID = " & CustomerID & _
    " And A.Agency = '" & AgencyCode & "' And A.CaseNumber = '" & CaseNumber & _
    "' And A.Episode = " & Episode & _
    " And (FinishDate IS NULL Or FinishDate = #12:00:00 PM#)" & _
    " And A.InterventionID = IC.Code", cnn, adOpenForwardOnly, adLockReadOnly
```

```
If rstCheck.RecordCount > 0 Then
tmpBuf(v) = vbNewLine & "There is a Computer Assessment that is not complete"
End If
```

```
rstCheck.Close
```

```
w = 0
v = v + 1
```

```
rstCheck.Open "Select Distinct Description From EmploymentAssessment A, InterventionCodes IC " & _
    "Where A.ConsumerID = " & CustomerID & _
    " And A.Agency = '" & AgencyCode & "' And A.CaseNumber = '" & CaseNumber & _
    "' And A.Episode = " & Episode & _
    " And (FinishDate IS NULL Or FinishDate = #12:00:00 PM#)" & _
    " And A.InterventionID = IC.Code", cnn, adOpenForwardOnly, adLockReadOnly
```

```
If rstCheck.RecordCount > 0 Then
tmpBuf(v) = vbNewLine & "There is a Employment Assessment that is not complete"
Check_Close_Episode = True
End If
```

```
rstCheck.Close
```

```
w = 0
v = v + 1
```

```
rstCheck.Open "Select Distinct Description From ILAssessment A, InterventionCodes IC " & _
    "Where A.ConsumerID = " & CustomerID & _
    " And A.Agency = '" & AgencyCode & "' And A.CaseNumber = '" & CaseNumber & _
    "' And A.Episode = " & Episode & _
    " And (FinishDate IS NULL Or FinishDate = #12:00:00 PM#)" & _
    " And A.InterventionID = IC.Code", cnn, adOpenForwardOnly, adLockReadOnly
```

```
If rstCheck.RecordCount > 0 Then
tmpBuf(v) = vbNewLine & "There is a IL Assessment that is not complete"
Check_Close_Episode = True
End If
```

```
rstCheck.Close
```

```
w = 0
v = v + 1
```

```
rstCheck.Open "Select Distinct Description From SocialAssessment A, InterventionCodes IC " & _
    "Where A.ConsumerID = " & CustomerID & _
    " And A.Agency = '" & AgencyCode & "' And A.CaseNumber = '" & CaseNumber & _
    "' And A.Episode = " & Episode & _
    " And (FinishDate IS NULL Or FinishDate = #12:00:00 PM#)" & _
    " And A.InterventionID = IC.Code", cnn, adOpenForwardOnly, adLockReadOnly
```

```
If rstCheck.RecordCount > 0 Then
tmpBuf(v) = vbNewLine & "There is a Social Work Assessment that is not complete"
End If
```

```
rstCheck.Close
```

```
' All Interventions must be closed
```

```
w = 0
```



v = v + 1

```
rstCheck.Open "Select Distinct Description From Interventions A, InterventionCodes IC " & _
    "Where A.ConsumerID = " & CustomerID & _
    " And A.Agency = '" & AgencyCode & "' And A.CaseNumber = '" & CaseNumber & _
    " And A.Episode = '" & Episode & "' _
    " And (EndDate IS NULL Or EndDate = #12:00:00 PM#)" & _
    " And A.InterventionID = IC.Code", cnn, adOpenForwardOnly, adLockReadOnly
```

If rstCheck.RecordCount > 0 Then

With rstCheck

While Not (.EOF)

If w > 0 Then

tmpBuf(v) = tmpBuf(v) & " and "

End If

tmpBuf(v) = tmpBuf(v) & ![Description]

.MoveNext

w = w + 1

Wend

End With

tmpBuf(v) = vbNewLine & "There are " & tmpBuf(v) & " Interventions that are not completed"

Check\_Close\_Episode = True

End If

rstCheck.Close

If Check\_Close\_Episode = True Then

For w = 0 To 9

tmpMessage = tmpMessage & tmpBuf(w)

Next

MsgBox tmpMessage, vbOKOnly, "Mandatory Items"

End If

Exit Function

errSave:

MsgBox Err.Description, vbCritical + vbOKOnly, "Check\_Close\_Episode"

End Function

Private Sub txtTriage\_Change(Index As Integer)

If Index = 6 Or Index = 9 Or Index = 13 Or Index = 17 Then

If Len(txtTriage(Index)) = 4 Then

If Not IsNumeric(Mid(txtTriage(Index), 2, 3)) Then

MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, " Phone Number"

txtTriage(Index) = Mid(txtTriage(Index), 1, 2)

txtTriage(Index).SelStart = 2

Else

txtTriage(Index) = txtTriage(Index) & "-"

txtTriage(Index).SelStart = 7

End If

ElseIf Len(txtTriage(Index)) = 9 Then

If Not IsNumeric(Mid(txtTriage(Index), 7, 3)) Then

MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, " Phone Number"

txtTriage(Index) = Mid(txtTriage(Index), 1, 6)

txtTriage(Index).SelStart = 7

Else

txtTriage(Index) = txtTriage(Index) & "--"

txtTriage(Index).SelStart = 11

End If

ElseIf Len(txtTriage(Index)) = 14 Then

If Not IsNumeric(Mid(txtTriage(Index), 11, 3)) Then

MsgBox "Please enter numeric number for phone number!", vbInformation + vbOKOnly, " Phone Number"

```
txtTriage(Index) = Mid(txtTriage(Index), 1, 10)
txtTriage(Index).SetFocus
End If
End If
End Sub

Private Sub txtTriage_Click(Index As Integer)
    If Index = 19 Then
        If (Check_Close_Episode = True) Then
            txtTriage(19).Locked = True
        Else
            txtTriage(19).Locked = False
        End If
    End If
End Sub

Private Sub txtTriage_GotFocus(Index As Integer)

If Index = 6 Or Index = 9 Or Index = 13 Or Index = 17 Then
    If txtTriage(Index) = "" Then
        txtTriage(Index) = "("
        txtTriage(Index).SelStart = 2
    End If
ElseIf Index = 19 Then
    If (Check_Close_Episode = True) Then
        txtTriage(19).Locked = True
    Else
        txtTriage(19).Locked = False
    End If
End If

End Sub

Private Sub txtTriage_KeyUp(Index As Integer, KeyCode As Integer, Shift As Integer)

If Index = 6 Or Index = 9 Or Index = 13 Or Index = 17 Then
    If KeyCode = vbKeyBack Then
        txtTriage(Index) = "("
        txtTriage(Index).SelStart = 2
    End If
End If

End Sub

Private Sub txtTriage_LostFocus(Index As Integer)
Dim intAge As Long
Dim today As Date

today = Now

If Index = 0 Then
    If IsDate(txtTriage(Index)) Then
        intAge = DateDiff("yyyy", txtTriage(Index), today)

        If today < DateSerial(Year(today), Month(txtTriage(Index)), Day(txtTriage(Index))) Then i
Age = intAge - 1
        txtTriage(20) = intAge
    End If
ElseIf Index = 1 Then
    If CheckForShiftTab Then
        cboYesNo(2).SetFocus
    ElseIf cboYesNo(2) = "No" Then
        tabTriage.Tab = 1
    End If
ElseIf Index = 2 Then
    If cboYesNo(2).ListIndex = 1 Then txtTriage(Index) = ""
ElseIf Index = 4 Then
    If CheckForShiftTab Then
        If cboYesNo(9).Text = "No" Then
            cboYesNo(9).SetFocus
        End If
    End If
End If
```

```

    ElseIf cboYesNo(9).Text = "Yes" And 1stHealthCondition.Selected(28) Then
        txtTriage(3).SetFocus
    ElseIf cboYesNo(9).Text = "Yes" And Not 1stHealthCondition.Selected(28) Then
        1stHealthCondition.SetFocus
    End If
End If
ElseIf Index = 6 Or Index = 13 Or Index = 17 Then
    If Trim(txtTriage(Index)) = "" Then txtTriage(Index) = ""
ElseIf Index = 9 Then
    If Trim(txtTriage(Index)) = "" Then txtTriage(Index) = ""
    If CheckForTab Then tabTriage.Tab = 2
ElseIf Index = 10 Then
    If CheckForTab Then tabTriage.Tab = 6
ElseIf Index = 13 Then
    If CheckForTab Then
        If cboAcceptReject(1).Text <> "Accept" Then txtTriage(15).SetFocus
    End If
ElseIf Index = 19 Then
    If CheckForTab Then tabTriage.Tab = 7
End If
End Sub

Private Sub txtTriage_Validate(Index As Integer, Cancel As Boolean)

If Index = 0 And Trim(txtTriage(0)) <> "" Then
    If Not IsDate(txtTriage(0)) Then
        MsgBox "Please enter your birth day!", vbInformation + vbOKOnly, "Birth Day"
        txtTriage(0) = ""
        Cancel = True
    End If
ElseIf Index = 1 Then
    If Trim(txtTriage(1)) <> "" Then
        If Not IsNumeric(txtTriage(1)) Then
            MsgBox "Please enter your age when you lost vision into this field!", vbInformation +
OKOnly, "Age at onset"
            Cancel = True
        End If
    End If
ElseIf Index = 2 Then
    If Trim(txtTriage(2)) <> "" Then
        If Not IsDate(txtTriage(2)) Then
            MsgBox "Please enter date when you last see your doctor!", vbInformation + vbOKOnly, "
test Date to see eye doctor "
            Cancel = True
        End If
    End If
ElseIf Index = 14 Then
    If Trim(txtTriage(14)) <> "" Then
        If Not IsDate(txtTriage(14)) Then
            MsgBox "Please enter date for your eye surgery or laser treatments!", vbInformation +
OKOnly, "Eye Surgery"
            Cancel = True
        End If
    End If
End If
End Sub

Private Function getTabCaption(tabIndex As Integer) As String

Select Case tabIndex
Case 0
    getTabCaption = "A. Demographics"
Case 1
    getTabCaption = "B. Health"
Case 2
    getTabCaption = "C. Problem ID"
Case 3
    getTabCaption = "D. Problem ID"
Case 4
    getTabCaption = "E. Risk Factors"

```

```
Case 5
  getTabCaption = "F. Recommendations"
Case 6
  getTabCaption = "G. Eye Report"
Case 7
  getTabCaption = "7-OB"
End Select

End Function

Private Function AcceptRecommendation() As Boolean
Dim i As Integer

For i = 0 To cboAcceptReject.Count - 1
  If cboAcceptReject(i).Text = "Accepted" Then
    AcceptRecommendation = True
    Exit For
  End If
Next
End Function
```

```
Option Explicit
Private IsNewRecord As Boolean
Private SequenceNo As Long
Private InterventionSequence As Long
Private rstVisit As ADODB.Recordset
```

```
Private Sub cmdClose_Click()
Unload Me
End Sub
```

```
Private Sub cmdNew_Click()
```

```
clearData cboStaffs, False
clearData cboServiceUnitType, False
clearData cboServiceMode, False
clearData cboServiceSession, False
clearData cboDuration, False
txtServiceDate = ""
```

```
IsNewRecord = True
cmdSave.Enabled = True
End Sub
```

```
Private Sub cmdNext_Click()
Dim i As Integer
Dim j As Integer
```

```
SaveInfoToDB
```

```
With rstVisit
.MoveNext
If Not (.EOF) Then
If ![ServiceDate] = "12:00:00 PM" Then
txtServiceDate = ""
Else
txtServiceDate = ![ServiceDate]
End If
SequenceNo = ![VisitsSeq]
LoadDataToControls rstVisit, cboStaffs, False
LoadDataToControls rstVisit, cboServiceUnitType, False
LoadDataToControls rstVisit, cboServiceMode, False
LoadDataToControls rstVisit, cboServiceSession, False
LoadDataToControls rstVisit, cboDuration, False
```

```
txtServiceDate.SetFocus
txtCurrent = .AbsolutePosition
```

```
If .RecordCount > 1 Then
If .AbsolutePosition = 1 Then
cmdPrevious.Enabled = False
cmdNext.Enabled = True
ElseIf .AbsolutePosition < .RecordCount Then
cmdPrevious.Enabled = True
cmdNext.Enabled = True
ElseIf .AbsolutePosition = .RecordCount Then
cmdPrevious.Enabled = True
cmdNext.Enabled = False
End If
End If
```

```
End With
```

```
End Sub
```

```
Private Sub cmdPrevious_Click()
Dim i As Integer
Dim j As Integer
```

```
SaveInfoToDB
```

```

With rstVisit
.MovePrevious
If Not (.BOF) Then
If ![ServiceDate] = "12:00:00 PM" Then
txtServiceDate = ""
Else
txtServiceDate = ![ServiceDate]
End If
SequenceNo = ![VisitsSeq]
LoadDataToControls rstVisit, cboStaffs, False
LoadDataToControls rstVisit, cboServiceUnitType, False
LoadDataToControls rstVisit, cboServiceMode, False
LoadDataToControls rstVisit, cboServiceSession, False
LoadDataToControls rstVisit, cboDuration, False

txtServiceDate.SetFocus
txtCurrent = .AbsolutePosition

If .RecordCount > 1 Then
If .AbsolutePosition = 1 Then
cmdPrevious.Enabled = False
cmdNext.Enabled = True
ElseIf .AbsolutePosition < .RecordCount Then
cmdPrevious.Enabled = True
cmdNext.Enabled = True
ElseIf .AbsolutePosition = .RecordCount Then
cmdPrevious.Enabled = True
cmdNext.Enabled = False
End If
End If
End With
End Sub

Private Sub cmdSave_Click()

SaveInfoToDB
MsgBox "Successfully save the information!", vbOKOnly, "Save Info"
End Sub

Private Sub Form_Load()
Dim i As Integer, j As Integer
Dim IndexNo As Integer

On Error GoTo Errhandler

txtFirstName = FirstName
txtLastName = LastName
txtCaseNumber = CaseNumber

txtIntervention = frmIntervention1.txtIntervention
InterventionSequence = frmIntervention1.SequenceNo
LoadControlsItems cboStaffs, False
LoadControlsItems cboServiceUnitType, False
LoadControlsItems cboServiceMode, False
LoadControlsItems cboServiceSession, False
LoadControlsItems cboDuration, False

Set rstVisit = New ADODB.Recordset
rstVisit.CursorLocation = adUseClient

rstVisit.Open "Select * From Visits Where ConsumerID=" & CustomerID & " And Agency = '" & _
AgencyCode & "' And CaseNumber = '" & CaseNumber & "' And InterventionID = '" & _
frmIntervention1.InterventionCode & "' And InterventionSequence = '" & InterventionSequence & _
" Order by VisitsSeq DESC", cnn, adOpenDynamic, adLockOptimistic
With rstVisit
txtTotal = .RecordCount
If Not (.BOF And .EOF) Then
If .RecordCount > 1 Then cmdNext.Enabled = True
If ![ServiceDate] = "12:00:00 PM" Or Trim(![ServiceDate] & vbNullString) = "" Then

```

```
txtServiceDate = ""
Else
    txtServiceDate = ![ServiceDate]
End If
SequenceNo = ![VisitsSeq]
LoadDataToControls rstVisit, cboStaffs, False
LoadDataToControls rstVisit, cboServiceUnitType, False
LoadDataToControls rstVisit, cboServiceMode, False
LoadDataToControls rstVisit, cboServiceSession, False
LoadDataToControls rstVisit, cboDuration, False
End If

If .AbsolutePosition < 0 Then
    txtCurrent = 0
Else
    txtCurrent = .AbsolutePosition
End If

End With

Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Form Load"
End Sub

Private Sub Form_Unload(Cancel As Integer)
'Dim response As Integer

    'response = MsgBox("Would you like save the Visit information ?", vbQuestion + vbYesNo, "C
")
    'If response = vbYes Then
        'MsgBox "Please use Save button to save it!", vbInformation + vbOKOnly, "Close"
        'Cancel = True
    'Else
        If rstVisit.State = adStateOpen Then rstVisit.Close
        Set rstVisit = Nothing
        Set frmVisits1 = Nothing
    'End If
End Sub

Private Sub SaveInfoToDB()
Dim tmpDate As Date
Dim tmpStaff As String
Dim tmpType As String
Dim tmpMode As String
Dim tmpSession As String
Dim tmpDuration As String
Dim tmpPosition As Long

On Error GoTo Errhandler

If cboStaffs.ListIndex = -1 Then
    tmpStaff = ""
Else
    tmpStaff = cboStaffs.ItemData(cboStaffs.ListIndex)
End If

If cboServiceUnitType.ListIndex = -1 Then
    tmpType = ""
Else
    tmpType = cboServiceUnitType.ItemData(cboServiceUnitType.ListIndex)
End If

If cboServiceSession.ListIndex = -1 Then
    tmpSession = ""
Else
    tmpSession = cboServiceSession.ItemData(cboServiceSession.ListIndex)
End If
```

```

If cboServiceMode.ListIndex = -1 Then
    tmpMode = ""
Else
    tmpMode = cboServiceMode.ItemData(cboServiceMode.ListIndex)
End If

If cboDuration.ListIndex = -1 Then
    tmpDuration = ""
Else
    tmpDuration = cboDuration.ItemData(cboDuration.ListIndex)
End If

If Trim(txtServiceDate) = "" Then
    tmpDate = "12:00:00 PM"
Else
    tmpDate = txtServiceDate.Text
End If

If (rstVisit.BOF And rstVisit.EOF) Or IsNewRecord Then
    cnn.Execute "INSERT INTO Visits (ConsumerID, Agency, Episode, CaseNumber, InterventionID, " & _
        " InterventionSequence, ServiceDate, StaffName, Type, Mode, Duration, SessionType) VA" & _
        S (" & CustomerID & ", ' " & _
            AgencyCode & ", ' " & Episode & ", ' " & CaseNumber & ", ' " & _
            frmIntervention1.InterventionCode & ", ' " & InterventionSequence & ", ' " & tmpDate & _
            " & tmpStaff & ", ' " & tmpType & ", ' " & _
            tmpMode & ", ' " & tmpDuration & ", ' " & tmpSession & "')"

    rstVisit.Requery
    SequenceNo = rstVisit![VisitsSeq]
    txtCurrent = rstVisit.AbsolutePosition
    txtTotal = rstVisit.RecordCount
    IsNewRecord = False

    With rstVisit
        If .RecordCount > 1 Then
            If .AbsolutePosition = 1 Then
                cmdPrevious.Enabled = False
                cmdNext.Enabled = True
            ElseIf .AbsolutePosition < .RecordCount Then
                cmdPrevious.Enabled = True
                cmdNext.Enabled = True
            ElseIf .AbsolutePosition = .RecordCount Then
                cmdPrevious.Enabled = True
                cmdNext.Enabled = False
            End If
        End If
    End With
Else
    cnn.Execute "Update Visits Set StaffName = '" & tmpStaff & "', Type = '" & tmpType & "', M" & _
        "= '" & _
        tmpMode & "', SessionType = '" & tmpSession & "', Duration = '" & tmpDuration & _
        "', ServiceDate = '" & tmpDate & "' Where VisitsSeq = " & SequenceNo
    tmpPosition = rstVisit.AbsolutePosition
    rstVisit.Requery
    rstVisit.AbsolutePosition = tmpPosition
End If
Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save Infor to DB"
End Sub

Private Sub txtServiceDate_Validate(Cancel As Boolean)

If Trim(txtServiceDate) = "" Then
    Cancel = False
ElseIf Not IsDate(txtServiceDate) Then
    MsgBox "Please enter date for Service Date!", vbInformation + vbOKOnly, "ServiceDate"
    Cancel = True

```



Visits1 - 5

60439708 .0113

End If  
End Sub

Dim Str, response As String

Private Sub cmdClose\_Click()  
    Unload Me  
End Sub

Private Sub cmdHints\_Click()  
Dim response As Integer

Str = "Here is a list of shortcut keys to help you navigate more quickly through this applicati  
" & vbNewLine & vbNewLine & \_  
    "Press Alt S to save a screen" & vbNewLine & \_  
    "Press Alt C to close a screen" & vbNewLine & \_  
    "Press Alt P to go to the Previous Page" & vbNewLine & \_  
    "Press Alt N to go to the Next Page"  
response = MsgBox(Str, vbOKOnly, "")  
End Sub

Private Sub cmdSearch\_Click()  
    frmSearchMenu.Show  
    Unload Me  
End Sub

Private Sub cmWelcom\_Click()  
Str = "Welcome to the Electronic Vision Rehabilitation Record" & vbNewLine & \_  
    "Copyright 2002 Lighthouse International" & vbNewLine & \_  
    "Property of Lighthouse International" & vbNewLine & \_  
    "Unauthorized use is prohibited" & vbNewLine & vbNewLine & \_  
    "Press the OK button to continue then press Alt H for listing of shortcut keys"  
  
response = MsgBox(Str, vbOKOnly, "Welcome")  
End Sub

Private Sub Form\_Activate()  
Me.Refresh  
End Sub

Private Sub Form\_Load()

Set cnn = New ADODB.Connection  
cnn.CursorLocation = adUseClient  
cnn.Open "DSN=miniEVR" & " " & App.Minor & " " & App.Revision

lblVersion = App.Major & "." & App.Minor & "." & App.Revision  
End Sub

```

Public CustomerID As Long
Public AgencyCode As String
Public Address As String
Public PhoneNo As String
Public FirstName As String
Public LastName As String
Public Episode As Long
Public CaseNumber As String
Public cnn As ADODB.Connection
Public ILSequenceNo As Long
Public SQLString As String
Public DiagnosisArray(1017) As String
Public DiagnosisString(26) As String
Public VisualFieldArray(11) As String
Public Declare Function GetKeyboardState Lib "user32" (pbKeyState As Byte) As Long

```

```
Public Function CheckForTab() As Boolean
```

```

    Const VK_TAB = &H9
    Const VK_SHIFT = &H10
    Dim btArray(255) As Byte

```

```
    Call GetKeyboardState(btArray(0))
```

```

    If ((btArray(VK_SHIFT) And &H80) <> &H80) And ((btArray(VK_TAB) And &H80) = &H80) Then
        CheckForTab = True
    End If

```

```
End Function
```

```
Public Function CheckForShiftTab() As Boolean
```

```

    Const VK_TAB = &H9
    Const VK_SHIFT = &H10
    Dim btArray(255) As Byte

```

```
    Call GetKeyboardState(btArray(0))
```

```

    If ((btArray(VK_SHIFT) And &H80) = &H80) And ((btArray(VK_TAB) And &H80) = &H80) Then
        CheckForShiftTab = True
    End If

```

```
End Function
```

```

Public Sub LoadControlsItems(ByVal tmpControls As Object, IsControlArray) 'Just add items to
ch ComboBox and ListBox
    Dim i As Integer
    Dim IndexNo As Integer
    Dim tmpRst As New ADODB.Recordset

```

```
On Err GoTo Errhandler
```

```

If IsControlArray Then 'for control array cases
    tmpRst.Open "Select * From " & Mid(tmpControls(0).Name, 4), cnn, adOpenForwardOnly, adLockRe
Only

```

```
    For i = 0 To tmpControls.Count - 1
```

```
        With tmpRst
```

```
            .MoveFirst
```

```
            While (Not .EOF)
```

```
                If tmpControls(0).Name = "cboDiagnosis" Then
```

```
                    tmpControls(i).ItemData(tmpControls(i).NewIndex) = IndexNo
```

```
                    DiagnosisArray(IndexNo) = ![Code]
```

```
                    IndexNo = IndexNo + 1
```

```
                Else
```

```
                    tmpControls(i).AddItem .Fields.Item("description")
```

```
'Add item to comb
```

```
or list Box
```

```
                    tmpControls(i).ItemData(tmpControls(i).NewIndex) = .Fields.Item("Code")
```

```
                End If
```

```
            .MoveNext
```

```
        Wend
```

```
    End With

```

```

Next
Else
    If tmpControls.Name = "cboStaffs" Then 'for single combobox or listbox cases
        tmpRst.Open "Select * From " & Mid(tmpControls.Name, 4) & " Order by StaffName", cnn, adOpenForwardOnly, adLockReadOnly
    Else
        tmpRst.Open "Select * From " & Mid(tmpControls.Name, 4), cnn, adOpenForwardOnly, adLockReadOnly
    End If

    With tmpRst
        .MoveFirst
        While (Not .EOF)
            If tmpControls.Name = "cboStaffs" Then
                tmpControls.AddItem .Fields.Item("StaffName") 'Add item to combo or list Box
                tmpControls.ItemData(tmpControls.NewIndex) = .Fields.Item("ID")
            Else
                tmpControls.AddItem .Fields.Item("Description")
                tmpControls.ItemData(tmpControls.NewIndex) = .Fields.Item("Code")
            End If
            .MoveNext
        Wend
    End With
    tmpRst.Close
    Set rsttmp = Nothing
    Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Load Control Items"
End Sub

Public Sub LoadDataToControls(ByVal tmpRst As Recordset, ByVal tmpControls As Object, IsArray As Boolean)
    Dim i As Integer
    Dim j As Integer
    Dim k As Integer
    Dim tmpStr As String

    On Error GoTo Errhandler

    With tmpRst
        If IsArray Then
            If TypeOf tmpControls(0) Is TextBox Then
                For i = 0 To tmpControls.Count - 1
                    'Debug.Print i
                    If tmpControls(i).Tag = "FinishDate" Or tmpControls(i).Tag = "AssessmentDate" Or _
                        tmpControls(i).Tag = "PriorDate" Or tmpControls(i).Tag = "RegistrationDate" Or _
                        tmpControls(i).Tag = "LastEyeExam" Or tmpControls(i).Tag = "EpisodeCloseDate" Or _
                        tmpControls(i).Tag = "EyeSurgeryDate" Or tmpControls(i).Tag = "PeriodFrom" Or _
                        tmpControls(i).Tag = "PeriodTo" Or tmpControls(i).Tag = "Birthdate" Or _
                        tmpControls(i).Tag = "LeavingTime1" Then
                        If .Fields.Item(tmpControls(i).Tag) = "12:00:00 PM" Or IsNull(.Fields.Item(tmpControls(i).Tag)) Then
                            tmpControls(i).Text = ""
                        Else
                            tmpControls(i).Text = .Fields.Item(tmpControls(i).Tag).Value & vbNullString
                        End If
                    Else
                        tmpControls(i).Text = .Fields.Item(tmpControls(i).Tag).Value & vbNullString
                    End If
                Next
            ElseIf TypeOf tmpControls(0) Is CheckBox Then
                For i = 0 To tmpControls.Count - 1
                    If Trim(.Fields.Item(tmpControls(i).Tag).Value & vbNullString) = "" Then
                        tmpControls(i).Value = 0
                    Else
                        tmpControls(i).Value = .Fields.Item(tmpControls(i).Tag).Value
                    End If
                Next
            End If
        End With
    End Sub

```

```

ElseIf TypeOf tmpControls(0) Is ComboBox Then
    For i = 0 To tmpControls.Count - 1
        If tmpControls(0).Name = "cboDiagnosis" Then
            If Trim(.Fields.Item(tmpControls(i).Tag) & vbNullString) = "" Then
                tmpControls(i).ListIndex = -1
            Else
                For j = 0 To UBound(DiagnosisArray)
                    If DiagnosisArray(j) = .Fields.Item(tmpControls(i).Tag) Then
                        k = j
                        Exit For
                    End If
                Next
                tmpControls(i).ListIndex = k
            End If
        ElseIf tmpControls(0).Name = "cboVisualField" Then
            If Trim(.Fields.Item(tmpControls(i).Tag) & vbNullString) = "" Then
                tmpControls(i).ListIndex = -1
            Else
                For j = 0 To UBound(VisualFieldArray)
                    If VisualFieldArray(j) = .Fields.Item(tmpControls(i).Tag) Then
                        k = j
                        Exit For
                    End If
                Next
                tmpControls(i).ListIndex = k
            End If
        Else
            If Trim(.Fields.Item(tmpControls(i).Tag) & vbNullString) = "" Then
                tmpControls(i).ListIndex = -1
            Else
                For j = 0 To tmpControls(i).ListCount - 1
                    If tmpControls(i).ItemData(j) = CLng(.Fields.Item(tmpControls(i).Tag).Value) Then
                        k = j
                        Exit For
                    End If
                Next
                tmpControls(i).ListIndex = k
            End If
        End If
    Next
ElseIf TypeOf tmpControls(0) Is ListBox Then
    For i = 0 To tmpControls.Count - 1
        tmpStr = .Fields.Item(tmpControls(i).Tag) & vbNullString
        For j = 1 To Len(tmpStr)
            If Mid(tmpStr, j, 1) = "1" Then
                tmpControls(i).Selected(j - 1) = True
            Else
                tmpControls(i).Selected(j - 1) = False
            End If
        Next
    Next
    tmpStr = ""
End If
Else
    If TypeOf tmpControls Is ListBox Then
        tmpStr = .Fields.Item(tmpControls.Tag) & vbNullString
        For i = 1 To Len(tmpStr)
            If Mid(tmpStr, i, 1) = "1" Then
                tmpControls.Selected(i - 1) = True
            Else
                tmpControls.Selected(i - 1) = False
            End If
        Next
    ElseIf TypeOf tmpControls Is TextBox Then
        If tmpControls.Tag = "CompletionDate" Or tmpControls.Tag = "AuthStartDate" Or _
            tmpControls.Tag = "AuthEndDate" Then
            If .Fields.Item(tmpControls.Tag) = "12:00:00 PM" Or IsNull(.Fields.Item(tmpControls
ag)) Then
                tmpControls = ""
            End If
        End If
    End If
End If

```

```

Else
    tmpControls = .Fields.Item(tmpControls.Tag).Value
End If
Else
    tmpControls = .Fields.Item(tmpControls.Tag).Value & vbNullString 'Ken add VBNUL
ng 8/9/02
End If
Else
    If .Fields.Item(tmpControls.Tag) & vbNullString = "" Then
        tmpControls.ListIndex = -1
    Else
        For i = 0 To tmpControls.ListCount - 1
            If tmpControls.ItemData(i) = CLng(.Fields.Item(tmpControls.Tag).Value) Then
                j = i
                Exit For
            End If
        Next
        tmpControls.ListIndex = j
    End If
End If
End With
Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Load data to Controls"
End Sub

Public Sub clearData(ByVal tmpControls As Object, IsArray As Boolean)
Dim i As Integer

If IsArray Then
    'for control array case
    If TypeOf tmpControls(0) Is TextBox Then
        For i = 0 To tmpControls.Count - 1
            tmpControls(i).Text = ""
        Next
    ElseIf TypeOf tmpControls(0) Is ComboBox Then
        For i = 0 To tmpControls.Count - 1
            tmpControls(i).ListIndex = -1
        Next
    ElseIf TypeOf tmpControls(0) Is CheckBox Then
        For i = 0 To tmpControls.Count - 1
            tmpControls(i).Value = 0
        Next
    End If
Else
    'for single control case
    If TypeOf tmpControls Is ListBox Then
        For i = 0 To tmpControls.ListCount - 1
            tmpControls.Selected(i) = False
        Next
    Else
        tmpControls.ListIndex = -1
    End If
End If
End Sub

Public Sub GoalObjectives(ByVal tmpForm As Form, InterventionID As String, SequenceNo As Long)
Dim i As Integer
Dim SQLStr As String
Dim GoalVar As Long
Dim tmpVar As Long
Dim ObjectiveVar As Long
Dim rst As New ADODB.Recordset

'get GoalID and ObjectiveID by tagName
On Error GoTo Errhandler

For i = 0 To tmpForm.cboAgree.Count - 1
    If tmpForm.cboAgree(i).Text = "Yes" Then
        rst.Open "Select GoalID, ObjectiveID From Objectives Where AliasName =" & tmpForm.cboAgree

```

```

(i).Tag & " ", _
    cnn, adOpenForwardOnly, adLockReadOnly
    GoalVar = rst![GoalID]
    ObjectiveVar = rst![ObjectiveID]
    rst.Close

    rst.Open "Select * From ProgressReport Where ConsumerID=" & CustomerID & " And Episode=
Episode & _
    " And InterventionID = " & InterventionID & " And AssessmentSequence = " & Sequence
    _
    cnn, adOpenForwardOnly, adLockReadOnly
    If rst.EOF And rst.EOF Then 'check the record existence
        SQLStr = "Insert into ProgressReport(ConsumerID,Agency,Episode,CaseNumber,Interventi
        " & _
        "AssessmentSequence,EntryDate) Values(" & CustomerID & "," & AgencyCode & "," &
sode & _
        " & CaseNumber & "," & InterventionID & "," & SequenceNo & ",Now())"
        cnn.Execute SQLStr
        rst.Requery 'requery the rst recordset to get the ProgressSeq No. used by next
ordset.
        End If
        tmpVar = rst![ProgressSeq]
        rst.Close

        rst.Open "Select * From GoalAchieve Where ConsumerID=" & CustomerID & " And Episode= "
isode & _
        " And InterventionID = " & InterventionID & " And ProgressSeq = " & tmpVar & _
        " And GoalID = " & GoalVar, cnn, adOpenForwardOnly, adLockReadOnly
        If rst.EOF And rst.EOF Then
            SQLStr = "Insert into GoalAchieve(ConsumerID,Agency,Episode,CaseNumber,Intervention:
ogressSeq," & _
            "GoalID,EntryDate) Values(" & CustomerID & "," & AgencyCode & "," & Episode &
& CaseNumber & _
            " & InterventionID & "," & tmpVar & "," & GoalVar & ",Now())"
            cnn.Execute SQLStr
            rst.Requery
            End If
            tmpVar = rst![GoalSeq]
            rst.Close

            rst.Open "Select * From ObjectiveAchieve Where ConsumerID=" & CustomerID & " And Episo
& Episode & _
            " And InterventionID = " & InterventionID & " And GoalSeq = " & tmpVar & " And Go
= " & _
            GoalVar & " And ObjectiveID = " & ObjectiveVar, cnn, adOpenForwardOnly, adLockReadC
            If rst.EOF And rst.EOF Then
                SQLStr = "Insert into ObjectiveAchieve(ConsumerID,Agency,Episode,CaseNumber,Interve
ID,GoalSeq," & _
                "GoalID,ObjectiveID,EntryDate) Values(" & CustomerID & "," & AgencyCode & "," & E
e & "," & _
                CaseNumber & "," & InterventionID & "," & tmpVar & "," & GoalVar & "," & Objecti
& ",Now())"
                cnn.Execute SQLStr
                End If
                rst.Close
            End If
        Next
    Exit Sub

Errhandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Goals & Objectives"
End Sub

Public Function SaveDataToDB(ByVal tmpForm As Form, ByVal tmpRst As Recordset, ByVal tmpCol
llection, InterventionCode As String, Is2ndTable As Boolean) As Boolean
    Dim i As Integer
    Dim j As Integer
    Dim tmpStr As String
    Dim IndexNo As Long
    Dim rst As ADODB.Recordset

```

On Error GoTo Errhandler

```

With tmpRst
  If (.EOF And .BOF) Or tmpForm.IsNewRecord Then
    .AddNew
    .Fields.Item("ConsumerID") = CustomerID
    .Fields.Item("Agency") = AgencyCode
    .Fields.Item("Casenumber") = CaseNumber
    If tmpForm.Name <> "frmTriage" And tmpForm.Name <> "frmEyeData" Then
      .Fields.Item("Episode") = Episode
      .Fields.Item("InterventionID") = InterventionCode
    End If
    If tmpForm.Name = "frmILAssessment" And Is2ndTable Then .Fields.Item("AssessmentSequence") = ILSequenceNo
  End If

  For i = 1 To tmpCol.Count
    'Debug.Print i
    If TypeOf tmpCol(i) Is ListBox Or TypeOf tmpCol(i) Is ComboBox Or TypeOf tmpCol(i) Is TextBox Then
      'for single control case
      If TypeOf tmpCol(i) Is ListBox Then
        For j = 0 To tmpCol(i).ListCount - 1
          If tmpCol(i).Selected(j) Then
            tmpStr = tmpStr & "1"
          Else
            tmpStr = tmpStr & "0"
          End If
        Next
        .Fields.Item(tmpCol(i).Tag) = tmpStr
        tmpStr = ""
      ElseIf TypeOf tmpCol(i) Is TextBox Then
        .Fields.Item(tmpCol(i).Tag) = tmpCol(i).Text
      Else
        If tmpCol(i).ListIndex = -1 Then
          .Fields.Item(tmpCol(i).Tag) = ""
        Else
          .Fields.Item(tmpCol(i).Tag) = tmpCol(i).ItemData(tmpCol(i).ListIndex)
        End If
      End If
      .Update
    Else
      'for control collection case
      If TypeOf tmpCol(i)(0) Is CheckBox Then
        For j = 0 To tmpCol(i).Count - 1
          .Fields.Item(tmpCol(i)(j).Tag) = tmpCol(i)(j).Value
        Next
        .Update
      ElseIf TypeOf tmpCol(i)(0) Is TextBox Then
        For j = 0 To tmpCol(i).Count - 1
          Debug.Print j
          If tmpCol(i)(j).Tag = "FinishDate" Or tmpCol(i)(j).Tag = "LeavingTime1" Or tmpCol(i)(j).Tag = "AssessmentDate" Or tmpCol(i)(j).Tag = "PriorDate" Or tmpCol(i)(j).Tag = "LastEyeExam" Or tmpCol(i)(j).Tag = "EyeSurgeryDate" Or tmpCol(i)(j).Tag = "EpisodeCloseDate" Then
            If Trim(tmpCol(i)(j)) = "" Then
              .Fields.Item(tmpCol(i)(j).Tag) = "12:00:00 PM"
            Else
              .Fields.Item(tmpCol(i)(j).Tag) = CDate(tmpCol(i)(j))
            End If
          Else
            .Fields.Item(tmpCol(i)(j).Tag) = Trim(tmpCol(i)(j))
          End If
        Next
        .Update
      ElseIf TypeOf tmpCol(i)(0) Is ListBox Then
        For j = 0 To tmpCol(i).Count - 1
          For k = 0 To tmpCol(i)(j).ListCount - 1
            If tmpCol(i)(j).Selected(k) Then
              tmpStr = tmpStr & "1"
            Else
              tmpStr = tmpStr & "0"
            End If
          Next k
        Next j
        .Fields.Item(tmpCol(i).Tag) = tmpStr
      End If
    End If
  Next
End With

```



```

        tmpStr = tmpStr & "0"
    End If
Next
.Fields.Item(tmpCol(i)(j).Tag) = tmpStr
tmpStr = ""
Next
Else
    For j = 0 To tmpCol(i).Count - 1
        'Debug.Print j
        If tmpCol(i)(0).Name = "cboDiagnosis" Then
            If tmpCol(i)(j).ListIndex = -1 Then
                .Fields.Item(tmpCol(i)(j).Tag) = ""
            Else
                IndexNo = tmpCol(i)(j).ItemData(tmpCol(i)(j).ListIndex)
                .Fields.Item(tmpCol(i)(j).Tag) = DiagnosisArray(IndexNo)
            End If
        ElseIf tmpCol(i)(0).Name = "cboVisualField" Then
            If tmpCol(i)(j).ListIndex = -1 Then
                .Fields.Item(tmpCol(i)(j).Tag) = ""
            Else
                IndexNo = tmpCol(i)(j).ItemData(tmpCol(i)(j).ListIndex)
                .Fields.Item(tmpCol(i)(j).Tag) = VisualFieldArray(IndexNo)
            End If
        Else
            If tmpCol(i)(j).ListIndex = -1 Then
                .Fields.Item(tmpCol(i)(j).Tag) = ""
            Else
                .Fields.Item(tmpCol(i)(j).Tag) = tmpCol(i)(j).ItemData(tmpCol(i)(j).List
(x)
                End If
            End If
        Next
        .Update
    End If
End If
Next
.Requery
If Not Is2ndTable Then
    If tmpForm.Name <> "frmTriage" Then
        tmpForm.txtCurrentRecord = .AbsolutePosition
    Else
        tmpForm.txtEpisode = ![Episode]
    End If
    If ![FinishDate] = "12:00:00 PM" Then
        'If filedate empty, save "12:00:00 PM"
        fault
        tmpForm.cmdSave.Enabled = True
        tmpForm.cmdClear.Enabled = False
    Else
        'call Goal/Objective and lock Save But
        If (tmpForm.Name <> "frmTriage" And tmpForm.Name <> "frmEyeData") Then
            GoalObjectives tmpForm, InterventionCode, ![AssessmentSequence]
            tmpForm.IsFileDateChecked = True
            tmpForm.cmdPrint.Enabled = True
        End If
        tmpForm.cmdSave.Enabled = False
        tmpForm.cmdClear.Enabled = True
    End If
    tmpForm.txtRecordNo = .RecordCount
    If .RecordCount > 1 Then
        If .AbsolutePosition = 1 Then
            tmpForm.cmdPreviousRecord.Enabled = False
            tmpForm.cmdNextRecord.Enabled = True
        ElseIf .AbsolutePosition < .RecordCount Then
            tmpForm.cmdPreviousRecord.Enabled = True
            tmpForm.cmdNextRecord.Enabled = True
        ElseIf .RecordCount = .AbsolutePosition Then
            tmpForm.cmdPreviousRecord.Enabled = True
            tmpForm.cmdNextRecord.Enabled = False
        End If
    End If

```

Module1 - 8

60439708.0113!

```
End If
End If
End If
End With
SaveDataToDB = True
Exit Function
```

```
ErrorHandler:
MsgBox Err.Description, vbCritical + vbOKOnly, "Save to DB"
SaveDataToDB = False
End Function
```

- The operation and interface of additional facets of the present invention, including regulatory compliance, billing, calendar, and additional analysis and reporting is substantially similar to those listed above and will be apparent to service providers and other users of the present invention. Additionally, although the present invention and
- 5 preferred embodiments have been described in considerable detail, modifications and other versions are possible and have been contemplated in the appended claim. Therefore, the spirit and scope of the appended claim should not be limited to the description of the preferred versions contained herein.

## CLAIMS

I claim:

- 5 A method of facilitating the management of consumer care, services and data, the method comprising: storing clinical pathway templates, said template being a pre-determined consumer care and service path; assigning a template to a given consumer upon initial data entry; deciding at least one clinical pathway at a data point for said consumer corresponding to a decision point within said template; tailoring said template at least
- 10 once adjusting for the consumers needs within predetermined paths; displaying said adjusted template on a screen reflecting consumer needs; storing templates entered relative to the consumer needs; storing data entered into said templates; providing a modified consumer pre-determined template; providing access to said stored data and templates; manipulating stored data and templates to display consumer goals and
- 15 objectives; in the form of progress providing an appointment scheduler; providing detailed functional questions in assessment areas; providing uniform forms; providing standardization of service quality with said forms and analyses; providing consistency of interviews with scripts, prompts and reminders; providing informed consent data and HIPAA requirement data; manipulation of data and templates to provide consumer
- 20 improvement; manipulation of data and templates to provide service provider improvement; manipulation of data and templates to provide clinical pathway method improvement; manipulate data to provide reports; provide quality analysis; provide billing computations; provide electronic billing external parties; provide corrective actions; provide a plurality of protection levels for consumer data and templates;
- 25 displaying requested data on at least one computer screen; recording all changes made to consumer templates.

## ABSTRACT

A method and system for vision rehabilitation services consumer care is provided. Clinical pathways and templates are selected based on consumer data entered and may be tailored to the consumer's needs and requirements. Consumer data is securely exchanged via the Internet between a central computer and consumer care provider computer to enable the method and system to perform program analysis, aid in and perform evaluations, track data, generate reports, standardize care, provide security, schedule appointments, monitor quality control, record and measure progress and variances, coordinate billing and payment, and reduce administrative cost.

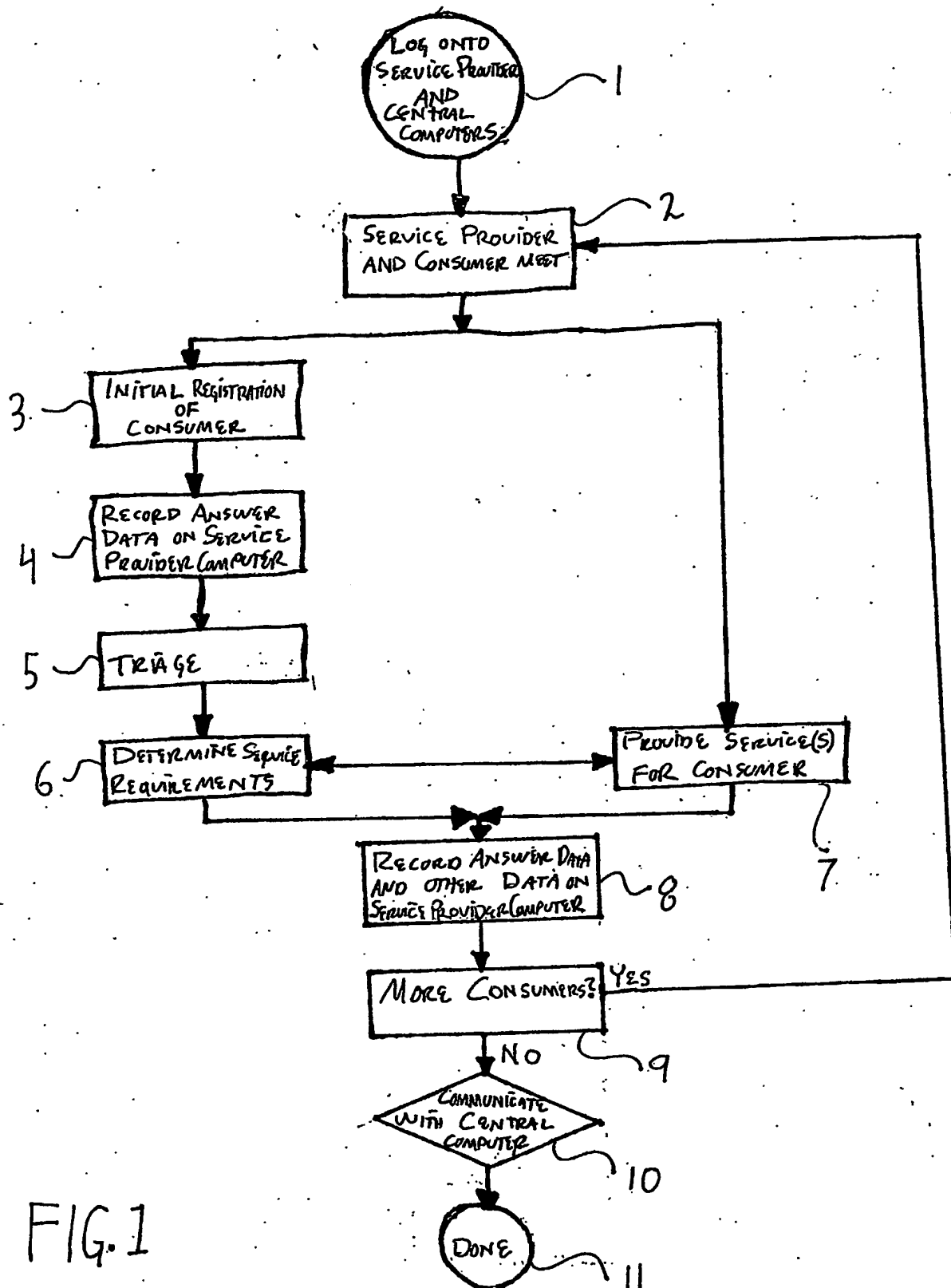


FIG. 1

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☒ **BLACK BORDERS**

☒ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

☒ **FADED TEXT OR DRAWING**

☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**

☐ **SKEWED/SLANTED IMAGES**

☒ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**

☐ **GRAY SCALE DOCUMENTS**

☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**

☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**